MetaDefender ICAP Server 4.7.3
# Table of Contents

## About This Guide

8

## About MetaDefender ICAP Server

9

### Summary

9

### ICAP

9

### MetaDefender ICAP Server

10

- Web Gateway or NGFW Integration
- Web Proxy Integration
- Reverse Proxy Integration
- Web Proxy Servers Supported

## Feedback

13

## 1. Quick Start with MetaDefender ICAP Server

14

### 1.1 Installation

14

- Installing MetaDefender ICAP Server on Ubuntu or Debian computers
- Installing MetaDefender ICAP Server on Red Hat Enterprise Linux or CentOS computers

15

- Installing MetaDefender ICAP Server on Windows

15

### 1.1.1 Basic configuration wizard

15

### 1.2 License activation

21

### 1.3 Creating MetaDefender Core server profile

23

- Creating Core server profile

23

### 1.4 Basic security rule

24

- Prerequisites

24

- Configuration

25

### 1.5 Web traffic redirection

25

- Example

25
2. Installing or Upgrading MetaDefender ICAP Server

2.1. Before Installation
   2.1.1 Increase capacity and resiliency
   2.1.2 System Requirements
   2.1.3 Sizing guide

2.2 Installing MetaDefender ICAP Server
   Installation overview
   Installation
   Installation notes
   2.2.1 Installing MetaDefender ICAP Server using the command line

2.3. Upgrading MetaDefender ICAP Server
   Notes for upgrading from MetaDefender ICAP Server v3
   Upgrading from MetaDefender ICAP Server v4

2.4. MetaDefender ICAP Server Licensing
   2.4.1 Activating MetaDefender ICAP Server licenses
   2.4.2 Checking your MetaDefender ICAP Server license

3. Configuring MetaDefender ICAP Server

3.1. MetaDefender ICAP Server configuration
   3.1.1 Web Management Console
   3.1.2 MetaDefender ICAP Server configuration file

3.2 Configuring TLS
   Web Management Console
   ICAP interface

3.3 User management
   Users and groups tab
   Roles tab
   User directories tab
   Transport Layer Security
   LDAP attributes
   Active Directory attributes
   Notes
   3.3.1 Change user password
   3.3.2 Active Directory attributes
   3.3.3 LDAP attributes
4.4.4 McAfee Web Gateway 205
4.4.5 Squid 209
4.4.6 FortiGate 213
4.4.7 A10 SSL Insight (SSLi) 218

4.5 Other Integration 228
   Next Generation Firewalls: 228

4.6 HTTP Header Handling 229
   Filename 229

4.7 Data trickling 232
   The problem 233
   Solution: data trickling 233

4.8 Risks with range request 238

5. Troubleshooting MetaDefender ICAP Server 240

5.1 How to create support package 240
   Linux 240
   Windows 241
   Content of the created package 241

5.2 How to read the MetaDefender ICAP Server log 242
   Linux 242
   Windows 243

5.3 Inaccessible Web Management Console 243
   Connection refused or browser is waiting 243
   Invalid license 244

5.4 Startup failure 245
   Could not create temporary directory 245

5.5 Can't view scan details 245
   Symptoms 245

6. Release Notes 248

   Previous releases 250

6.1 Changelog 250
   Version 4.7.3 251
How long is the support life cycle for a specific version/release of MetaDefender ICAP Server v4? 270

How to upgrade from ICAP Server 4.0.0 to a newer ICAP Server v4 release 271

Installing ICAP Server on Debian 9 273
   Symptom 273
   Reason 273
   Resolution 273
About This Guide

This guide is intended to provide the information you need to:

- Install, configure, and manage MetaDefender ICAP Server v4.x.

⚠️ If you are using MetaDefender ICAP Server v3.x, refer to MetaDefender ICAP Server v3.x user guide.

- Learn about new features, updated features, and bug fixes on each MetaDefender ICAP Server release (i.e. each product version’s release notes).
- Learn about frequently asked questions and additional concepts through our library of knowledge base articles.

While we offer the option to download this guide to a PDF file, it is optimized for online browser viewing. OPSWAT updates the online version of the guide regularly on an "as needed" basis. By viewing the document online, you are assured that you are always seeing the most recent and most comprehensive version of the guide.
About MetaDefender ICAP Server

Summary

The MetaDefender ICAP Server product is designed to enable scanning the contents of HTTP traffic—that enter or leave an internal network— for advanced threats.

MetaDefender ICAP Server provides ICAP interface between MetaDefender Core and ICAP clients (mainly Proxies: Web or Reverse Proxy Servers). Any content routed through the ICAP interface will be scanned with the same anti-malware engines and policies as files scanned through any other MetaDefender Core interface. Scan results can be cached by MetaDefender Core and/or by the Proxy, which can significantly improve scanning throughput and lower traffic load.

Scanning with a MetaDefender ICAP Server also allows logging files entering the network. This log information may be used later as evidence during the investigation of security incidents.

ICAP

The Internet Content Adaption Protocol (ICAP) is, in essence, a lightweight protocol for executing a "remote procedure call" on HTTP messages. It allows ICAP clients to pass HTTP messages to ICAP servers for some sort of transformation or other processing ("adaptation"). The server executes its transformation service on messages and sends back responses to the client, usually with modified messages. Typically, the adapted messages are either HTTP requests or HTTP responses. [IETF]

ICAP is a lightweight HTTP-like protocol specified in RFC 3507 which is used to extend transparent proxy servers, thereby freeing up resources and standardizing the way in which new features are implemented. ICAP is generally used to implement virus scanning and content filters in transparent HTTP proxy caches. Content adaptation refers to performing the particular value added service (content manipulation) for the associated client request /response. [WIKI]

ICAP concentrates on leveraging edge-based devices (caching proxies) to help deliver value-added services. At the core of this process is a cache that will proxy all client transactions and will process them through ICAP web servers. These ICAP servers are focused on a specific function, for example malware scanning. Off-loading value-added services from web servers to ICAP servers allows those same web servers to be scaled according to raw HTTP throughput versus having to handle these extra tasks. [WIKI]
MetaDefender ICAP Server

MetaDefender ICAP Server provides the ICAP interface on top of MetaDefender Core.

When a user uploads data over HTTP (for example with a PUT or POST request), the contents of the request are forwarded to MetaDefender Core by MetaDefender ICAP Server for scanning. When a user downloads data from an external server (for example with a GET request), the contents of the reply are also forwarded for scanning, before being sent to the user’s computer. HTTP requests and responses are always redirected to the MetaDefender ICAP server, independently from the HTTP method.

Based on the scan results MetaDefender ICAP Server may either accept or reject the request:

- If attached data is clean, then MetaDefender ICAP Server accepts the traffic and returns HTTP contents that can be forwarded normally by the Proxy. Based on the configuration of MetaDefender Core, clean files can be sanitized, so the contents of a clean file may still be modified.
- If attached data is identified as a threat, then MetaDefender ICAP Server rejects the traffic and modifies the HTTP request or reply accordingly (e.g., a custom HTML message): the original, malicious content won’t reach its intended destination.

Web Gateway or NGFW Integration

MetaDefender ICAP Server allows system administrators to easily integrate MetaDefender Core’s multi-scanning technology into an existing web gateway or new-generation firewall to enable anti-malware scanning of all HTTP downloads and uploads. Any web gateway or new-generation firewall that implements ICAP – such as Fortinet FortiGate – can be set up to automatically forward HTTP requests to MetaDefender ICAP Server.
Web Proxy Integration
MetaDefender ICAP Server allows system administrators to easily integrate MetaDefender Core’s multi-scanning technology into an existing web proxy to enable anti-malware scanning of all HTTP downloads and uploads. Any proxy that implements ICAP –such as Blue Coat® ProxySG or Squid– can be set up to automatically forward HTTP requests to MetaDefender ICAP Server.

Reverse Proxy Integration
MetaDefender ICAP Server allows system administrators to easily integrate MetaDefender Core’s multi-scanning technology into an existing reverse proxy to enable anti-malware scanning of all HTTP file uploads. Any reverse proxy that implements ICAP –such as F5® BIG-IP® Load Traffic Manager™ (LTM®)– can be set up to automatically forward any uploaded files to MetaDefender ICAP Server.
Web Proxy Servers Supported

- 4.4.1 ARA network JAGUAR5000
- 4.4.2 Symantec Blue Coat Proxy SG
- 4.4.3 F5 Integrations
- 4.4.4 McAfee Web Gateway
- 4.4.5 Squid
- 4.4.6 FortiGate
- 4.4.7 A10 SSL Insight (SSLi)
Feedback

For comments and questions regarding this document, please contact OPSWAT on the Support tab at https://portal.opswat.com/.
1. Quick Start with MetaDefender ICAP Server

This guide describes the basic steps for installing and using MetaDefender ICAP Server:

- 1.1 Installation
- 1.2 License activation
- 1.3 Creating MetaDefender Core server profile
- 1.4 Basic security rule
- 1.5 Web traffic redirection

This Quick Guide assumes that the test machine has working Internet connection.

1.1 Installation

Before starting the installation please make sure your server computer or virtual machine meets the minimum hardware and software requirements.

Installing MetaDefender ICAP Server on Ubuntu or Debian computers

1. Download *mdicapsrv* package from the OPSWAT Portal. Make sure that you download the applicable package for your distribution.

2. Upload the installation package to your server computers.

3. Install the product with the following command (**<filename>** is the MetaDefender ICAP Server package you downloaded from our portal):

   ```
   $ sudo dpkg -i <filename>
   ```

4. If dpkg shows error messages about missing dependencies you should execute:

   ```
   $ sudo apt-get install -f
   ```

5. Open a web browser and point to (**<server name or IP>** is the DNS name or IP address of your server):

   ```
   http://<server name or IP>:8048
   ```
6. Enter default login credentials, username: \texttt{admin}, password: \texttt{admin}

**Installing MetaDefender ICAP Server on Red Hat Enterprise Linux or CentOS computers**

1. Download \texttt{mdicapsrv} package from the OPSWAT Portal. Make sure that you download the applicable package for your distribution
2. Upload the installation package to your server computers
3. Install the product with the following command (\texttt{<filename>} is the MetaDefender ICAP Server package you downloaded from our portal):

\[
\texttt{$ \text{sudo yum install <filename>}$}
\]

4. Open a web browser and point to (\texttt{<server name or IP>} is the DNS name or IP address of your server):

\[
\texttt{http://<server name or IP>:8048}
\]

5. Enter default login credentials, username: \texttt{admin}, password: \texttt{admin}

**Installing MetaDefender ICAP Server on Windows**

1. Download \texttt{mdicapsrv} Windows installer from the OPSWAT Portal
2. Upload the installation package to your server computers
3. Install the product with executing the installer
4. Open a web browser and point to (\texttt{<server name or IP>} is the DNS name or IP address of your server):

\[
\texttt{http://<server name or IP>:8048}
\]

5. Enter default login credentials, username: \texttt{admin}, password: \texttt{admin}

To continue the basic setup, follow the license activation instructions on License activation.
For more information on Installation procedures see Installing MetaDefender ICAP Server.

\textbf{1.1.1 Basic configuration wizard}
Introduction

When trying to access the Web Management Console for the first time, you are to complete a basic configuration wizard in order to be able to use the product. The Web Management Console will be available only after you have successfully finished this wizard.

To start the wizard click CONTINUE.

⚠️ Sensitive information

This wizard may transfer sensitive information over an unencrypted connection. Always use this wizard on a secure, closed network or localhost, and with care!
Basic configuration steps

End-User License Agreement

In the first page you can find the End-User License Agreement. You have to accept the terms before moving on. Please read through the EULA carefully and if you agree with it, check I ACCEPT THE TERMS IN THE LICENSE AGREEMENT and click NEXT to continue.

Admin User Setup

The next step is to set up an administrator account. This account will be the first one being able to access the Web Management Console and to create accounts for other users. You have to fill all fields in this page to be able to move forward. When you are done, click NEXT to continue.

User directory

The administrator account, that is created via the basic configuration wizard, is always added to the LOCAL user directory as a member.

The following information is required for the administrator account:
<table>
<thead>
<tr>
<th>ACCOUNT NAME</th>
<th>The unique name of the account that is used at the time of login and in log messages for accountability.</th>
</tr>
</thead>
<tbody>
<tr>
<td>ACCOUNT DISPLAY NAME</td>
<td>Name of the person bound to this account. This name (appended to the name of the account's user directory) is displayed in the top right corner of the Web Management Console.</td>
</tr>
<tr>
<td>PASSWORD</td>
<td>Password of the user bound to this account that is used at the time of login.</td>
</tr>
<tr>
<td>EMAIL</td>
<td>Email address of the person bound to this account.</td>
</tr>
</tbody>
</table>

**Passwords sent clear-text**

As long as TLS is not configured for the basic configuration wizard, passwords are sent clear-text over the network and may be disclosed to unauthorized parties.

As a mitigation action:

1. Either use the wizard on *localhost* or on a direct network connection, or
2. Enable TLS as soon as possible and change the password immediately if it has already been set.
License activation

For license activation details see 2.4.1 Activating MetaDefender ICAP Server licenses.

Create Core Server Profile

In this step you can create a basic Core server profile which will be used for connecting to a MetaDefender Core instance so your traffic can be scanned and sanitized. You should give a name for the profile (e.g. "My MetaDefender Core"), set the address of your MetaDefender
Core instance and choose a rule from the list. In order to go to the next step the wizard will check if a connection can be made to the address provided. If the connection seems fine a NEXT button will appear in the place of the TEST button and you can continue the configuration process by clicking on it.

By skipping this step the wizard won't be able to create a security rule later however you will still be able to create a Core server profile and a security rule in the product after the wizard is finished. You will also have the chance to modify the profile created in the wizard with more advanced configuration possibilities later. For more information on server profiles please see 3.9 Server profiles.

Security Rule

If you have successfully finished creating a Core server profile your next step will be creating a security rule. In the wizard all you have to do is to choose a name for it and it will be generated using the previously created Core server profile as the one for scanning. You can have more advanced settings for it later, see 4.2 Security rules.

Click NEXT to continue the configuration process.

Wizard completion

After you have completed every steps you are ready to finish the wizard and start using the product. Click the FINISH button to complete the wizard.

⚠️ The product's service will be restarted and the browser will be redirected to the Web Management Console. This could take several seconds.

You can login to the Web Management Console with the administrator user that have just been created in the previous steps.
Transport Layer Security

Transport Layer Security (TLS) is a cryptographic protocol that provides communications security over a computer network. Websites, like the Web Management Console, are able to use TLS to secure all communications between their servers and web browsers.

The TLS protocol aims primarily to provide confidentiality (privacy) and data integrity between two communicating computer applications.

⚠️ No TLS for the wizard

By default, TLS is not enabled for the basic configuration wizard. As a consequence sessions between the wizard's backend and the browser may be insecure.

Performing the same steps as for the Web Management Console, it is possible to set up TLS for the basic configuration wizard. Remember completing the TLS setup before launching the wizard.

For instructions to set up TLS see 3.2 Configuring TLS.

1.2 License activation

To activate your installation go to the Settings > License menu in the Web Management Console. If you have no valid license, you will only see your installation's Deployment ID. You will also see a warning in the Web Management Console header.
1. Press the **ACTIVATE** button to bring up the Activation menu, where you should choose from the available modes:

   - **Online**: the product will contact the OPSWAT license server online, and acquire its license based on your Activation key and Deployment ID.
   - **Offline**: you can upload a manually acquired license file.
   - **Request trial key online**: if you want to try out the product first, you can receive a trial Activation key via email.

2. Select the desired option

3. Follow the on-screen instructions

4. Finally press the **SEND** button

After successful activation the product will be available with all its functionality. For detailed license activation instructions see **2.4.1 Activating MetaDefender ICAP Server Licenses**.

⚠️ When your hardware information changes, for example your mac address changes because the product runs in a virtual machine, the license get automatically reactivated on the first update attempt.
1.3 Creating MetaDefender Core server profile

After installation and successful license activation MetaDefender ICAP Server user interface is ready to be used with full functionality.

For requests to be scanned MetaDefender ICAP Server needs, however, to be connected to MetaDefender Core.

Establishing this connection requires two steps:

1. Creating a MetaDefender Core type server profile
2. Assigning this server profile to the ICAP Server (see 1.4 Basic security rule)

Creating Core server profile

Go to Inventory > Server profiles and click ADD NEW PROFILE. In the SERVER PROFILE TYPE drop-down list select MetaDefender Core and specify the details of your MetaDefender Core instance(s).

Properties not listed in the table below may be left on their default values or blank, or filled according to the organizational policies.

<table>
<thead>
<tr>
<th>Server profile type</th>
<th>Profile name</th>
<th>Server specifications (URI)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Value</td>
<td>MetaDefender Core</td>
<td>Core servers URI</td>
</tr>
<tr>
<td>Examples</td>
<td>MetaDefender Core</td>
<td>Core</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>URI example</th>
<th>Transport level encryption</th>
</tr>
</thead>
<tbody>
<tr>
<td><a href="http://10.0.0.10:8008">http://10.0.0.10:8008</a></td>
<td>None</td>
</tr>
<tr>
<td><a href="https://10.0.10:8008">https://10.0.10:8008</a></td>
<td>TLS</td>
</tr>
</tbody>
</table>
For further details about server profiles see 3.9 Server profiles.

1.4 Basic security rule

- **Prerequisites**
- **Configuration**

After installation, successful license activation and basic configuration MetaDefender ICAP Server is ready to be used.

However, MetaDefender ICAP Server blocks all requests by default. To allow web traffic some basic security rule configuration is needed.

⚠️ The basic security rules created in this chapter may be too permissive and may not be suitable for production deployments.

**Prerequisites**

The following information is needed to create the basic security rules:

1. MetaDefender Core type server profile containing the Core (see 1.3 Creating MetaDefender Core server profile)
Configuration

To allow web traffic go to Policy > Security rules and create the following basic security rule.

In the examples below we assume that
- The Core server is configured in the MetaDefender Core type serve profile CoreProfile.

Properties not listed in the table below may be left on their default values (if they have, e.g. settings on ACTIONS or ADVANCED tabs) or filled according to the organizational policies (e.g. USE TLS).

<table>
<thead>
<tr>
<th>Tab</th>
<th>SCAN</th>
</tr>
</thead>
<tbody>
<tr>
<td>Field</td>
<td>METADefeNDER CORE</td>
</tr>
<tr>
<td>Value</td>
<td>MetaDefender Core type server profile containing the Core</td>
</tr>
<tr>
<td>Examples</td>
<td>CoreProfile</td>
</tr>
</tbody>
</table>

For details about security rules in MetaDefender ICAP Server see 4.2 Security rules.

1.5 Web traffic redirection

As a final step –when all the previous steps in this chapter are done– the web traffic must be redirected to flow through MetaDefender ICAP Server.

MetaDefender ICAP Server's ICAP service is listening on port 1344 by default.

For further details about integrating MetaDefender ICAP Server with various ICAP clients see 4.4 Web Proxy Integrations and 4.5 Other Integration.

Example

In case of Squid proxy installed on the same machine as ICAP Server the following configuration must be appended to the squid.conf file:
Squid example

icap_enable on
icap_send_client_ip on
icap_preview_enable on
icap_preview_size 0
icap_service_failure_limit -1
icap_service metascan_req reqmod_precache bypass=0 icap://127.0.0.1:1344/OMSScanReq-AV
adaptation_access metascan_req allow all
icap_service metascan_resp respmod_precache bypass=0 icap://127.0.0.1:1344/OMSScanResp-AV
adaptation_access metascan_resp allow all
icap_persistent_connections on
2. Installing or Upgrading MetaDefender ICAP Server

- 2.1. Before Installation
- 2.2 Installing MetaDefender ICAP Server
- 2.3. Upgrading MetaDefender ICAP Server
- 2.4. MetaDefender ICAP Server Licensing

2.1. Before Installation

Before installing MetaDefender ICAP Server make sure the target computer meets the hardware and software requirements.

- 2.1.1 Increase capacity and resiliency
- 2.1.2 System Requirements
- 2.1.3 Sizing guide
2.1.1 Increase capacity and resiliency
### Sizing examples

<p>| | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Number of simultaneous users</strong></td>
<td>10</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Data throughput</strong> (kB/s)</td>
<td>8005</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Request throughput</strong> (req/s)</td>
<td>97.91280654</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Total requests</strong></td>
<td>35934</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Total time</strong></td>
<td>367</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>CPU cores</strong></td>
<td>2</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>RAM</strong></td>
<td>8 GB</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th><strong>Load</strong></th>
<th>Average</th>
<th>Minimum</th>
<th>Maximum</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>CPU (%)</strong></td>
<td>57</td>
<td>23</td>
<td>80</td>
</tr>
<tr>
<td><strong>IO (/SEC)</strong></td>
<td>877</td>
<td>377</td>
<td>2063</td>
</tr>
<tr>
<td><strong>RAM (BYTES)</strong></td>
<td>53 835 148</td>
<td>42 102 784</td>
<td>58 867 712</td>
</tr>
</tbody>
</table>

**Maximum possible values**

Please note that

1. Each CPU core can add 100% as a maximum, so the theoretical is 200%.
2. 8 GB is 8 589 934 592 bytes, that is the theoretical maximum for the RAM here.

Measurement methodology

1. MetaDefender Core connection was disabled to eliminate potential bottleneck caused by Core
2. 100 ms delay was applied between requests
3. Each user requested the whole dataset 10 times
   a. Files were requested in a randomized fashion

Test environment

Error rendering macro 'drawio' : null
Traffic mix

<table>
<thead>
<tr>
<th>File type</th>
<th>Share</th>
</tr>
</thead>
<tbody>
<tr>
<td>Archive (zip)</td>
<td>2.8 %</td>
</tr>
<tr>
<td>Image</td>
<td>2.5 %</td>
</tr>
<tr>
<td>Microsoft Office</td>
<td>10.7 %</td>
</tr>
<tr>
<td>PDF</td>
<td>1.5 %</td>
</tr>
<tr>
<td>Windows executable (.exe)</td>
<td>24.2 %</td>
</tr>
<tr>
<td>HTML</td>
<td>19.6 %</td>
</tr>
<tr>
<td>Text</td>
<td>37.3 %</td>
</tr>
<tr>
<td>Total</td>
<td>98.6 %</td>
</tr>
<tr>
<td>Rounding error</td>
<td>1.4 %</td>
</tr>
</tbody>
</table>

2.1.2 System Requirements

- Operating system and hardware requirements
- Other software requirements
  - MetaDefender Core
    - Version compatibility matrix
    - Security rules
  - Browser requirements
  - Additional installation of Windows services
- Ports that must be available
Operating system and hardware requirements

Please confirm that your system meets the minimum requirements listed below before installing MetaDefender ICAP Server.

- **Operating System:**
  - CentOS 6.6+, 7.0+
  - Red Hat Enterprise Linux 6.6+, 7.0+
  - Debian 8.0+, 9.0+
  - Ubuntu 16.04, 18.04
  - Windows 7+ (64 bit)
  - Microsoft Windows Server 2008 R2 or newer (64 bit)

- **Hardware requirements**
  - RAM: min. 2 GB
  - SSD: 2 GB + ~500MB * [number of managed scan engines]

**Performance concerns**

For performance reasons it is not recommended to use a HDD instead of SSD.

If MetaDefender Core or any other system is installed on the same physical machine as MetaDefender ICAP Server then the additional systems' hardware requirements need also be taken into consideration.
Other software requirements

MetaDefender Core

Version compatibility matrix

<table>
<thead>
<tr>
<th>MetaDefender ICAP Server</th>
<th>MetaDefender Core</th>
</tr>
</thead>
<tbody>
<tr>
<td>4.0.0</td>
<td>4.7.0+</td>
</tr>
<tr>
<td>4.1.0</td>
<td>4.8.0+</td>
</tr>
<tr>
<td>4.2.0+</td>
<td>4.8.2+</td>
</tr>
</tbody>
</table>

Scan engines

The following scan engines must be active:

<table>
<thead>
<tr>
<th>Scan engine</th>
<th>Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>Data sanitization</td>
<td>5.0.2-6 +</td>
</tr>
<tr>
<td>At least one of the available anti-malware engines</td>
<td>N/A</td>
</tr>
</tbody>
</table>

Security rules

Under Policy > Security rules on Core, security rules –that are used by MetaDefender ICAP Server’s security rules–

1. SHOULD enable the following Visibility of scan result:

<table>
<thead>
<tr>
<th>Role</th>
<th>Visibility</th>
</tr>
</thead>
<tbody>
<tr>
<td>Everybody</td>
<td>FULL DETAILS</td>
</tr>
</tbody>
</table>

⚠ Without this visibility ICAP Server can still work, but certain details may not be provided on the UI.
Browser requirements

One of the following desktop browsers is suggested to view the MetaDefender ICAP Server's UI:

- Chrome
- Firefox
- Safari
- Microsoft Edge
- Internet Explorer 11

Chrome, Firefox, Safari and Edge browsers are tested with the latest available version at the time of release.

⚠️ Mobile layouts are not supported yet.

Additional installation of Windows services

<table>
<thead>
<tr>
<th>Name</th>
<th>Service Name</th>
<th>Optional</th>
</tr>
</thead>
<tbody>
<tr>
<td>MetaDefender ICAP Server</td>
<td>mdicapsrv</td>
<td>REQUIRED</td>
</tr>
</tbody>
</table>

Ports that must be available

<table>
<thead>
<tr>
<th>Direction</th>
<th>Component / Service</th>
<th>Port</th>
<th>Note</th>
</tr>
</thead>
<tbody>
<tr>
<td>Inbound</td>
<td>MetaDefender ICAP Server</td>
<td>1344</td>
<td>Customizable; adjust accordingly if modified.</td>
</tr>
<tr>
<td>Inbound</td>
<td>MetaDefender ICAP Server</td>
<td>8048</td>
<td>Customizable; adjust accordingly if modified.</td>
</tr>
<tr>
<td>Outbound</td>
<td>MetaDefender Core</td>
<td>8008</td>
<td>only if MetaDefender Core is installed on a remote system.</td>
</tr>
</tbody>
</table>
### 2.1.3 Sizing guide

This document describes a subset of the configuration and sizing options for MetaDefender ICAP Server. If more detailed technical sizing is required, please contact your sales engineer or representative.

<table>
<thead>
<tr>
<th>Hardware</th>
<th>CPU cores</th>
<th>2</th>
<th>8</th>
<th>16</th>
</tr>
</thead>
<tbody>
<tr>
<td>RAM</td>
<td>4 GB</td>
<td>8 GB</td>
<td>16 GB</td>
<td></td>
</tr>
<tr>
<td>Storage</td>
<td>25 GB</td>
<td>50 GB</td>
<td>100 GB</td>
<td></td>
</tr>
<tr>
<td>NIC</td>
<td>1 GbE</td>
<td>1 GbE</td>
<td>1 GbE</td>
<td></td>
</tr>
<tr>
<td>Raw ICAP capacity*</td>
<td>Internet throughput (Mb/s)</td>
<td>100 Mb/s</td>
<td>250 Mb/s</td>
<td>500 Mb/s</td>
</tr>
<tr>
<td></td>
<td>Web traffic (req/s)</td>
<td>20</td>
<td>75</td>
<td>150</td>
</tr>
<tr>
<td></td>
<td>Employee count / Number of users</td>
<td>100</td>
<td>500</td>
<td>2500</td>
</tr>
<tr>
<td>ICAP + Core capacity**</td>
<td>Core servers / engine count</td>
<td>Bundled, Single / 4</td>
<td>Bundled, Single / 4</td>
<td>Separate, 2 x load balancing / 4</td>
</tr>
<tr>
<td></td>
<td>Internet throughput (Mb/s)</td>
<td>100 Mb/s</td>
<td>250 Mb/s</td>
<td>500 Mb/s</td>
</tr>
<tr>
<td></td>
<td>Web traffic (req/s)</td>
<td>12</td>
<td>20</td>
<td>50</td>
</tr>
<tr>
<td></td>
<td>Employee count / Number of users</td>
<td>20</td>
<td>100</td>
<td>500</td>
</tr>
</tbody>
</table>
The raw ICAP capacity is useful for cases when MetaDefender Core is going to be deployed to a separate server.

The ICAP + Core capacity is useful for cases when MetaDefender Core is going to be deployed to the same server as MetaDefender ICAP Server.

2.2 Installing MetaDefender ICAP Server

Installation overview
The following steps are needed to be executed to install MetaDefender ICAP Server

1. Download the package of your choice from the OPSWAT portal
2. Install the package on your computer via the Command Line
3. Open a web browser and point to (<server name or IP> is the DNS name or IP address of your test server):
   http://<server name or IP>:8048
4. Login with the default credentials, username: admin, password: admin
5. You must Activate this deployment to use its features

Installation

- 2.2.1 Installing MetaDefender ICAP Server using the command line

Installation notes

- If the MetaDefender ICAP Server package dependencies are not installed on your system you may need to have a working Internet connection or you may have to provide the Installation media during the installation. Consult your Operating System documentation on how to use Installation media as a package repository.
- During installation the databases might need to be upgraded. This could take noticeable time.
2.2.1 Installing MetaDefender ICAP Server using the command line

Preliminary notes

If the MetaDefender ICAP Server package dependencies are not installed on your system you may need to have a working Internet connection or you may have to provide the Installation media during the installation. Consult your Operating System documentation on how to use Installation media as a package repository.

For each case below start with the following steps:

1. Download *mdicapsrv* package from the [OPSWAT Portal](https://www.opswat.com/). Make sure that you download the applicable package for your distribution
2. Upload the installation package to your test computers
3. Install the product with the provided command (*<filename>* is the MetaDefender ICAP Server package you downloaded from our portal)

**Debian / Ubuntu package (.deb)**

```
$ sudo dpkg -i <filename> || sudo apt-get install -f
```

**Red Hat Enterprise Linux / CentOS package (.rpm)**

```
$ sudo yum install <filename>
```

**Windows package (.msi)**

On Windows systems it is possible to install the product by running the corresponding .msi file. From command line interface it is also possible to install the product by executing

```
> msiexec /i <filename> <option key>=<option value>
```

where the possible option keys and their default option values are the following:
<table>
<thead>
<tr>
<th>Key</th>
<th>Default Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>RESTADDRESS</td>
<td>0.0.0.0</td>
<td>REST interface binding address</td>
</tr>
<tr>
<td>RESTPORT</td>
<td>8048</td>
<td>REST interface binding port</td>
</tr>
<tr>
<td>ICAPADDRESS</td>
<td>0.0.0.0</td>
<td>ICAP interface binding address</td>
</tr>
<tr>
<td>ICAPPORT</td>
<td>1344</td>
<td>ICAP interface binding port</td>
</tr>
</tbody>
</table>

For details on using msiexec please consult Windows installer documentation.

2.3. Upgrading MetaDefender ICAP Server

**Notes for upgrading from MetaDefender ICAP Server v3**

⚠️ It is not possible to directly upgrade MetaDefender ICAP Server v3 to v4. MetaDefender ICAP Server v3 comes built into MetaDefender Core v3. Uninstall MetaDefender Core v3 first, then start a fresh installation.

⚠️ There is no support for importing MetaDefender ICAP Server v3 configuration into MetaDefender ICAP Server v4. The configuration must be migrated manually.

**Upgrading from MetaDefender ICAP Server v4**

To upgrade from a former version of MetaDefender ICAP Server v4 a simple installation of the latest version is enough.

All existing MetaDefender ICAP Server configuration and data will be kept during the upgrade.

2.4. MetaDefender ICAP Server Licensing

In order to use MetaDefender ICAP Server you need to activate the product.
If you already purchased ICAP server together with MetaDefender Core v3 then please contact OPSWAT sales for your activation key.

- 2.4.1 Activating MetaDefender ICAP Server licenses
- 2.4.2 Checking your MetaDefender ICAP Server license

2.4.1 Activating MetaDefender ICAP Server licenses

- Initial steps
- Online activation
- Offline activation
  - Offline activation details
- Request trial key online
  - Trial key request details
- Notes

Initial steps

1. To activate your installation go to the **Settings > License** menu in the Web Management Console. If you have no valid license, you will only see your installation’s Deployment ID. You will also see a warning in the Web Management Console header.
2. Press the *ACTIVATE* button to bring up the Activation menu. The following modes are available:

   a. Online
   b. Offline
   c. Request trial key online

**Online activation**

With internet connection on the server, the MetaDefender ICAP Server instance may be activated directly using the Activation key received at the time of purchasing the product.

**Offline activation**

With no internet connection on the server the MetaDefender ICAP Server instance may be activated indirectly from a different machine, that has internet connection. The Deployment ID of the MetaDefender ICAP Server instance and the Activation key received at the time of purchasing the product will be required. Follow the steps on the screen to activate the product offline.
Activation

**ACTIVATION MODE**

- ONLINE
- OFFLINE
- REQUEST TRIAL KEY ONLINE

Offline activation steps:

1. Copy down your Deployment ID: MICAP0NYMGTZjRxPH4y6jvJAPvR6VRevY1v7KoJ6
2. Go to OPSWAT portal: [https://portal.opswat.com/activation](https://portal.opswat.com/activation)
3. Activate and download your license file (you will need your Activation key and the Deployment ID of this instance)
4. Upload the license file here
5. Check license details in the license menu

**ACTIVATION FILE**

SEND  CANCEL
**Offline activation details**

<p>| | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Log on to <a href="https://portal.opswat.com/activation">https://portal.opswat.com/activation</a></td>
</tr>
<tr>
<td>2</td>
<td>Select <em>MetaDefender ICAP Server as MetaDefender Package</em></td>
</tr>
<tr>
<td>3</td>
<td>Fill in the requested information about your deployment</td>
</tr>
</tbody>
</table>
Click the Request Unlock Key button.
The Download Unlock Key link appears.

Metadefender Package

Metadefender ICAP Server

Activation Key *

Number Of End-User-Clients

10

Deployment ID *

MICAPGNYMCjT2jRxBMqGv36rVRsuV1s7KoNJ6

Optional Description

This helps you to identify this deployment OPSWAT License portal

Request Unlock Key

Please click this link to download license file.

Download Unlock Key
5. Click the Download Unlock Key link and save the activation file.

Request trial key online

An evaluation license may be acquired for 14 days. To obtain a trial key register on the OPSWAT Portal first.
### Trial key request details

1. **Provide your e-mail address (that was registered on the OPSWAT portal) in the **EMAIL ADDRESS** and click the **SEND** button.

2. **An e-mail response will arrive from sales@opswat.com containing your trial activation key and activation instructions.**

   ![E-mail example]

   - **Wed 2017-04-19 14:24**
   - **sales@opswat.com**
   - **[OPSWAT] Metadefender Core v4 Evaluation Key**
   - **To:** [Redacted]
   - **We removed extra line breaks from this message.**

   Thank you for requesting evaluation key for a Metadefender product!

   We registered a trial key request from your email address: [Redacted]

   Your activation key is 1f0C-g5U2-xpVe-AbWm-VMP3-4NyB-nRCh-zLub

   This key is valid for the following products:
   - Metadefender iCAP Server - Evaluation
   - [Redacted]

   To activate your Metadefender product installation, please follow the following:
   1. Access the web management console from a web browser.

3. **Using the received activation key continue with an **online** or **offline** activation.

### Notes

- **⚠️ If you activated your installation, but your license becomes invalid or expired, you will see a **RE-ACTIVATE** button.** After clicking it, the product tries to activate the license with the formerly entered activation information.
2.4.2 Checking your MetaDefender ICAP Server license

User menu pane

Basic license information is always visible in the user menu pane on the lower left side of the screen. The following information is available:

- Product version
- License expiration: last day of license validity

License menu

For more license details and activating your installation go to Settings > License menu on the Web Management Console:

- Product ID: product identification as on your order
- Product name: product name as on your order
- Expiration: last day of license validity
- Max clients: number of licenses committed / number of licenses allowed
- Deployment ID: identification of this installation
3. Configuring MetaDefender ICAP Server

3.1. MetaDefender ICAP Server configuration

3.1.1 Web Management Console

3.1.2 MetaDefender ICAP Server configuration file

3.2 Configuring TLS

3.3 User management

3.3.1 Change user password

3.3.2 Active Directory attributes

3.3.3 LDAP attributes

3.5 Logging

3.5.1 Configuration

3.5.2 Debug logging

3.5.3 Web Management Console logs

3.5.4 Logging traffic of bad requests

3.6. Internet connection

3.7 Other settings

3.8 Customizing the block page

3.9 Server profiles

3.9.1 MetaDefender Core specific inventory properties

3.1. MetaDefender ICAP Server configuration

The MetaDefender ICAP Server configuration is separated into two parts. The basic server configurations are stored in the configuration files. Other configuration values can be set via the Web Management Console.

3.1.1 Web Management Console

The Web Management Console is available at (<server name or IP> is the DNS name or IP address of your test server):
After installing the product the default password for the admin user is admin.

As long as TLS is not configured for the Web Management Console, passwords are sent clear-text over the network. To set up TLS see Configuring TLS.

Every change made in MetaDefender ICAP Server configuration via the Web Management Console is applied when you select Save settings or OK, except if the change cannot be applied.

Typical issues related to the Web Management Console:

- Inaccessible Web Management Console

3.1.2 MetaDefender ICAP Server configuration file

- Linux
  - [global] section
  - [logger] section
- Windows
Linux

⚠️ Configuration upgrades on RHEL/CentOS

When ICAP Server is upgraded on RHEL/CentOS, then the configuration file is not automatically upgraded if there were modifications to it.

In this case the installer (RPM) creates a file called `mdicapsrv.rpmnew` with the upgraded configuration entries, and this file needs to be merged manually to the actual configuration file.

The configuration file for the server is located in `/etc/mdicapsrv/mdicapsrv.conf`.

After modifying the server configuration file you must restart the MetaDefender ICAP Server service in order for the changes to take effect. You should use the distribution-standard way to restart the `mdicapsrv` service.

### [global] section

<table>
<thead>
<tr>
<th>parameter</th>
<th>default value</th>
<th>required</th>
<th>description</th>
</tr>
</thead>
<tbody>
<tr>
<td>icapaddress</td>
<td>0.0.0.0</td>
<td>required</td>
<td>One of the IP addresses of the computer that runs the product to serve ICAP interface (0.0.0.0 means all interface)</td>
</tr>
<tr>
<td>icapport</td>
<td>1344</td>
<td>required</td>
<td>Designated port number for the ICAP interface</td>
</tr>
<tr>
<td>restaddress</td>
<td>0.0.0.0</td>
<td>required</td>
<td>One of the IP addresses of the computer that runs the product to serve REST API and web user interface (0.0.0.0 means all interface)</td>
</tr>
<tr>
<td>restport</td>
<td>8048</td>
<td>required</td>
<td>Designated port number for the web and REST interface</td>
</tr>
<tr>
<td>parameter</td>
<td>default value</td>
<td>required</td>
<td>description</td>
</tr>
<tr>
<td>-------------------------------</td>
<td>--------------------------</td>
<td>----------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>tempdirectory</td>
<td>/var/tmp/mdicapsrv/temp</td>
<td>optional</td>
<td>Root directory for temporary files creation.</td>
</tr>
</tbody>
</table>
|                               |                          |          | ![Warning] A /temp subdirectory is automatically created within a customized directory. For example:  
|                               |                          |          | • If /tmp is configured as tempdirectory then  
|                               |                          |          | • /tmp/temp will be used for creating temporary files                        |
| skip_multipart_without_filename | false                   | optional | When enabled the ICAP server won't send files from a multipart request for scanning when the given part does not have a filename key in it's own Content-Disposition header |

**[logger] section**

<table>
<thead>
<tr>
<th>key</th>
<th>default value</th>
<th>required</th>
<th>description</th>
</tr>
</thead>
<tbody>
<tr>
<td>logfile</td>
<td>/var/log/mdicapsrv/mdicapsrv.log</td>
<td>optional</td>
<td>Full path of a logfile to write log messages to</td>
</tr>
<tr>
<td>loglevel</td>
<td>info</td>
<td>optional</td>
<td>Level of logging. Supported values are: debug, info, warning, error</td>
</tr>
<tr>
<td>key</td>
<td>default value</td>
<td>required</td>
<td>description</td>
</tr>
<tr>
<td>-------------------</td>
<td>---------------</td>
<td>------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>syslog</td>
<td>optional</td>
<td></td>
<td>Switch on logging to a local ('local') or remote ('protocol://hostname:port') syslog server. (Multiple server can be specified separated with comma)</td>
</tr>
<tr>
<td>syslog_level</td>
<td>optional</td>
<td></td>
<td>Level of logging. Supported values are: debug, info, warning, error</td>
</tr>
<tr>
<td>override</td>
<td>optional</td>
<td></td>
<td>Override specific log ids to display them on another level e.g.: &quot;1723:error,663:info&quot;. Note: when displaying these log ids their original level will remain the same.</td>
</tr>
<tr>
<td>capture_traffic</td>
<td>optional</td>
<td></td>
<td>Capture raw TCP traffic in case of bad requests. See 3.5.4 Logging traffic of bad requests.</td>
</tr>
<tr>
<td>cef</td>
<td>false</td>
<td>optional</td>
<td>If true, the log format is Common Event Format</td>
</tr>
<tr>
<td>local_timezone</td>
<td>false</td>
<td>optional</td>
<td>If true, the times sent in syslog messages will be in the server's local timezone. This does not effect entries in the log file/Windows event log. When syslog is used with cef and local_timezone enabled the timezone name can vary based on the underlying system and it's settings.</td>
</tr>
</tbody>
</table>

### Examples

- **Syslog**
  - UTC: 2018-09-19T13:07:36Z
  - Local: 2018-09-19T15:07:36+02:00
- **Syslog with CEF**
  - UTC: Sep 19 13:12:47 UTC
### Parameters

<table>
<thead>
<tr>
<th>key</th>
<th>default value</th>
<th>required</th>
<th>description</th>
</tr>
</thead>
<tbody>
<tr>
<td>nginx_logfile</td>
<td>/var/log/mdicapsrv/nginx-mdicapsrv.log</td>
<td>optional</td>
<td>File name and path to store the NGINX logs. If this value is changed, the /etc/logrotate.d/mdicapsrv should be changed accordingly.</td>
</tr>
</tbody>
</table>

⚠️ You should set both of syslog and syslog_level or none of them and you should set both of logfile and loglevel or none of them.

### Windows

The configuration for the server is located in **Windows Registry**.

After modifying the server configuration file you must restart the MetaDefender ICAP Server service in order for the changes to take effect.

Default logging target is Windows event log with default level of info (see below).

**HKEY_LOCAL_MACHINE\SOFTWARE\OPSWAT\ICAP Server\global**

<table>
<thead>
<tr>
<th>parameter</th>
<th>default value</th>
<th>type</th>
<th>required</th>
<th>description</th>
</tr>
</thead>
<tbody>
<tr>
<td>icapaddress</td>
<td>0.0.0.0</td>
<td>string value</td>
<td>required</td>
<td>One of the computer to serve ICAP interface. 0.0.0.0 means all</td>
</tr>
<tr>
<td>icapport</td>
<td>1344</td>
<td>string value</td>
<td>required</td>
<td>Designate ICAP interface port</td>
</tr>
</tbody>
</table>

- Local 1: Sep 19 15:12:47 CEST
- Local 2: Sep 19 15:12:47 Central Europe Daylight Time
<table>
<thead>
<tr>
<th>Parameter</th>
<th>Default Value</th>
<th>Type</th>
<th>Required</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>restaddress</td>
<td>0.0.0.0</td>
<td>string</td>
<td>required</td>
<td>One of the computer to serve REST API and web user interface (0.0.0.0 means all interface)</td>
</tr>
<tr>
<td>restport</td>
<td>8048</td>
<td>string</td>
<td>required</td>
<td>Designate port number for the web and REST interface</td>
</tr>
<tr>
<td>tempdirectory</td>
<td>C:\Program Files\OPSWAT\Metadefender ICAP Server\data\temp</td>
<td>string</td>
<td>optional</td>
<td>Root directory for temporary files creation. A subdirectory \temp is automatically created within a customized directory. For example:</td>
</tr>
<tr>
<td>skip_multipart_without_filename</td>
<td>false</td>
<td>string</td>
<td>optional</td>
<td>When enabled the ICAP server won't send files from a multipart request for scanning when the given part does not have a filename key in its own Content-Disposition header</td>
</tr>
</tbody>
</table>

HKEY_LOCAL_MACHINE\SOFTWARE\OPSWAT\ICAP Server\logger
<table>
<thead>
<tr>
<th>parameter</th>
<th>default value</th>
<th>type</th>
<th>required</th>
<th>description</th>
</tr>
</thead>
<tbody>
<tr>
<td>logfile</td>
<td></td>
<td>string value</td>
<td>optional</td>
<td>Location of a logfile to write log messages to</td>
</tr>
<tr>
<td>loglevel</td>
<td></td>
<td>string value</td>
<td>optional</td>
<td>Level of logging. Supported values are: debug, info, warning, error</td>
</tr>
<tr>
<td>wineventlog_level</td>
<td>info</td>
<td>string value</td>
<td>optional</td>
<td>Level of logging. Supported values are: debug, info, warning, error</td>
</tr>
<tr>
<td>syslog</td>
<td></td>
<td>string value</td>
<td>optional</td>
<td>Value can only by in form of 'udp://&lt;hostname&gt;:&lt;port&gt;'. (Multiple server can be specified separated with comma)</td>
</tr>
<tr>
<td>syslog_level</td>
<td></td>
<td>string value</td>
<td>optional</td>
<td>Level of logging. Supported values are: debug, info, warning, error</td>
</tr>
<tr>
<td>override</td>
<td></td>
<td>string value</td>
<td>optional</td>
<td>Override specific log ids to display them on another level e.g.: &quot;1723: error,663:info&quot;. Note: when displaying these log ids their original level will remain the same.</td>
</tr>
<tr>
<td>capture_traffic</td>
<td></td>
<td>DWORD</td>
<td>optional</td>
<td>Capture raw TCP traffic in case of bad requests. See 3.5.4 Logging traffic of bad requests.</td>
</tr>
<tr>
<td>cef</td>
<td>false</td>
<td>string value</td>
<td>optional</td>
<td>If true, the log format is Common Event Format</td>
</tr>
<tr>
<td>local_timezone</td>
<td>false</td>
<td>string value</td>
<td>optional</td>
<td>If true, the times sent in syslog messages will be in the server's local timezone. This does not effect entries in the log file/Windows event log.</td>
</tr>
<tr>
<td>parameter</td>
<td>default value</td>
<td>type</td>
<td>required</td>
<td>description</td>
</tr>
<tr>
<td>--------------</td>
<td>--------------</td>
<td>------</td>
<td>----------</td>
<td>------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>nginx logfile</td>
<td>[installdir]\nginx\nginx.log</td>
<td>string</td>
<td>optional</td>
<td>File name and path to store the NGINX logs. (Rotation of this log has not yet been solved on Windows systems)</td>
</tr>
</tbody>
</table>

When syslog is used with cef and local_timezone enabled the timezone name can vary based on the underlying system and it's settings.

Examples

- Syslog
  - UTC: 2018-09-19T13:07:36Z
  - Local: 2018-09-19T15:07:36+02:00

- Syslog with CEF
  - UTC: Sep 19 13:12:47 UTC
  - Local 1: Sep 19 15:12:47 CEST
  - Local 2: Sep 19 15:12:47 Central Europe Daylight Time
You should set both of syslog and syslog_level or none of them and you should set both of logfile and loglevel or none of them.

3.2 Configuring TLS

- Web Management Console
  - Linux
  - Windows
  - Important notes
- ICAP interface
  - Installation
    - Linux
      - Debian / Ubuntu (.deb)
      - Red Hat / CentOS (.rpm)
    - Windows
  - Configuration
    - Linux
      - Red Hat / CentOS
    - Windows
    - Important notes

⚠️ For production MetaDefender ICAP Server deployments a more sophisticated TLS configuration is recommended than what is described below. Please consult the nginx documentation on Configuring HTTPS servers and stunnel manual for further details.

⚠️ It is not recommended to use self-signed certificates in production environments. If you do not have a suitable certificate, you can apply to a Certificate Authority to obtain one.
Web Management Console

MetaDefender ICAP Server supports accessing Web Management Console via HTTPS. This feature is, however, not enabled by default. To enable the TLS/HTTPS you should modify MetaDefender ICAP Server configuration by following the next steps.

**Linux**

Let us assume that MetaDefender ICAP Server is installed in:

```
/path/to/mdicapsrv
```

and the server certificate is:

```
/path/to/certificate.crt
```

and the private key belonging to the certificate is:

```
/path/to/privatekey.key
```

To simply enable TLS:

1. Create file `ssl.conf` in the directory `/path/to/mdicapsrv/nginx.d`

2. Add the following lines:

```
ssl on;
ssl_certificate         /path/to/certificate.crt;
ssl_certificate_key     /path/to/privatekey.key;
```

3. Restart mdicapsrv service.

**Windows**

Let us assume that MetaDefender ICAP Server is installed in:

```
C:\Path\To\Metadefender ICAP Server
```

and the server certificate is:
C:\Path\To\certificate.crt

and the private key belonging to the certificate is:

C:\Path\To\privatekey.key

To simply enable TLS:

1. Create file ssl.conf in the directory C:\Path\To\Metadefender ICAP Serve\nginx.

2. Add the following lines:

   ```
   ssl on;
   ssl_certificate "C:\Path\To\certificate.crt";
   ssl_certificate_key "C:\Path\To\privatekey.key";
   ```


**Important notes**

⚠️ When choosing location for certificate and key files, make sure that these files are in a location which is readable to the service user.

Certificate and key files should be obtained and saved by the user in a convenient location, adjust the paths accordingly.

**ICAP interface**

MetaDefender ICAP Server does not support TLS protocol on the ICAP interface out of the box. Stunnel can be configured to accept ICAP requests from a TLS connection, decrypt the request, and pass it to the local MetaDefender ICAP Server.
Installation

Linux

Debian / Ubuntu (.deb)

$ sudo apt-get install stunnel

Red Hat / CentOS (.rpm)

$ sudo yum install stunnel

Windows

1. Download the Windows installer from stunnel's download page
2. Start the installer and follow its steps (use default values if you are not sure)
3. During the installation you will be asked to generate a self-signed certificate file. Fill in the required fields with your information
4. Make sure that "Start stunnel after installation" is not checked at the end of the setup

Configuration

Linux

Red Hat / CentOS

1. You need a valid certificate for stunnel regardless of what service you use it with. If you do not have a suitable certificate, you can apply to a Certificate Authority to obtain one, or you can create a self-signed certificate. To create the self-signed certificate for stunnel: [RHEL]

   # cd /etc/pki/tls/certs
   # make stunnel.pem
2. The certificate is being created. Fill in the required fields with your information.

3. Create and edit `/etc/stunnel/stunnel.conf` file and add the following lines:

```plaintext
[icaps]
accept = 11344
connect = 1344
cert = /etc/pki/tls/certs/stunnel.pem
```

Where:

a. `accept`: The port number where stunnel listens for TLS connections for the given service

b. `connect`: The port number where the decrypted connections are forwarded to. (This should be the port used by MetaDefender ICAP Server)

c. `cert`: The TLS certificate used by the service. You can set your own or use the one generated during stunnel setup (which is stunnel.pem next to stunnel.conf)

4. Save and close the configuration file.

5. Start stunnel with the following command:

```
# stunnel /etc/stunnel/stunnel.conf
```

**Windows**

Let us assume that stunnel is installed in:

```
C:\Path\To\stunnel
```

1. Locate and open the `stunnel.conf` file. It should be under the `config` directory in the stunnel installation directory. (e.g., `C:\Path\To\stunnel\config\stunnel.conf`)

2. Add the following lines at the end of the file:

```plaintext
[icaps]
accept = 11344
connect = 1344
cert = C:\Path\To\stunnel\config\stunnel.pem
```

Where:
a. **accept:** The port number where stunnel listens for TLS connections for the given service

b. **connect:** The port number where the decrypted connections are forwarded to. (This should be the port used by MetaDefender ICAP Server)

c. **cert:** The TLS certificate used by the service. You can set your own or use the one generated during stunnel setup (which is stunnel.pem next to stunnel.conf)

3. Save and close the configuration file.

4. Start stunnel service.

**Important notes**

⚠️ The certificate generated by stunnel is a self-signed certificate. It is not recommended to use self-signed certificates in production environments. If you do not have a suitable certificate, you can apply to a Certificate Authority to obtain one.

### 3.3 User management

- Users and groups tab
  - Legacy default user
  - Functions
    - Add new user from a Local type user directory
    - API keys
    - Add new users from an LDAP type or Active Directory type user directory
    - Add new group from an LDAP type or Active Directory type user directory
    - Assign roles to a user, LDAP group or Active Directory group
    - Delete user
  - Special user accounts
    - SYSTEM/management account

- Roles tab
  - Default roles
  - Permissions
    - Effective right
• Functions
  • Modify role
• User directories tab
  • Default user directories
  • Local type user directories
  • Active Directory type user directories
  • LDAP type user directories
• Functions
  • Add new Local type user directory
  • Add new LDAP type or Active Directory type user directory
    • Differences between LDAP and Active Directory type user directories
  • Delete user directory
  • Enable or disable user directory
  • Unlock locked accounts
• Transport Layer Security
• LDAP attributes
• Active Directory attributes
• Notes

To manage the users of the product, go to the Settings > User Management menu in the Web Management Console.

**Users and groups tab**

The USERS AND GROUPS tab lists the existing users, LDAP groups and Active Directory groups in the system.
Legacy default user

Previous versions of the product created a default user during the installation with the following credentials and parameters. This default user may still exist if the product was upgraded from previous versions.

<table>
<thead>
<tr>
<th>Username</th>
<th>Password</th>
<th>Name</th>
<th>Email</th>
<th>Roles</th>
<th>User directory</th>
</tr>
</thead>
<tbody>
<tr>
<td>admin</td>
<td>admin</td>
<td>Administrator</td>
<td>admin@localhost</td>
<td>Administrators</td>
<td>LOCAL</td>
</tr>
</tbody>
</table>

Functions

Besides listing existing users, LDAP and AD groups, the USERS AND GROUPS tab provides the following functions:

- Add new user, LDAP or AD group
- Modify (and view) existing user’s, LDAP or AD group’s properties
- Delete existing user, LDAP or AD group
Add new user from a Local type user directory

To add a new user from a Local type user directory click the ADD NEW USER button and select a Local type user directory in the USER DIRECTORY drop down list.

The field ASSIGN TO ROLES lists all the roles that are assigned to this user. See section Assign roles to a user, LDAP group or Active Directory group for details about role assignment.

⚠️ As long as TLS is not configured for the Web Management Console, passwords are sent clear-text over the network. For details see Transport Layer Security.

⚠️ If enhanced password policy is enabled for the user directory this user belongs to then the new password must fulfil the password complexity requirements listed at the Local type user directories section.

API keys

The APIKEY value provides access to the product's REST API for the user under editing with no authentication. If no such functionality is needed for the user then this field can be left blank.

There are two methods to create an APIKEY for a user:

1. Generate the APIKEY by using Generate link next to the APIKEY field,
2. Manually enter the APIKEY value; it must matches the following criteria:

### APIKEY validation criteria

- The length of the API key must be exactly 36 characters.
- It must contain numeric and lower case a, b, c, d, e and f letter characters only
  
  (e.g. "1x2y3z..." is invalid because of the x, y and z characters).
- It must contain at least 10 lower case a, b, c, d, e or f letter characters.
- It must contain at least 10 numeric characters.
- It is allowed to contain at most 3 consecutive lower case letter characters
  
  (e.g. "abcd1a2b3c..." is invalid because of the four consecutive letters).
- It is allowed to contain at most 3 consecutive numeric characters (e.g. "1234a1b2c3..." is invalid because of the four consecutive numeric characters).
Add new users from an LDAP type or Active Directory type user directory

To add a new user from an LDAP type or Active Directory type user directory click the ADD NEW USER button and select an LDAP type or Active Directory type user directory in the USER DIRECTORY drop down list. Select USER as the ACCOUNT TYPE.

Provide the name of the account in the ACCOUNT NAME field and click the FIND ACCOUNT button to look up the account in the LDAP or Active Directory. If the lookup succeeds then the ACCOUNT DISPLAY NAME and the DISTINGUISHED NAME fields are filled automatically.

Do provide the account name precisely. There is no functionality to look up similar names or partial matches.

The field ASSIGN TO ROLES lists all the roles that are assigned to this user. See section Assign roles to a user, LDAP group or Active Directory group for details about role assignment.
4.7.3

Add new group from an LDAP type or Active Directory type user directory

The purpose of adding an LDAP or Active Directory group to the product is to assign role(s) to all the users in that LDAP or Active Directory group.

The users of the LDAP or Active Directory group can authenticate with their LDAP or Active Directory credentials in the product's Web Management Console and will be assigned with the roles assigned to the group.

To add a new group from an LDAP type or Active Directory type user directory click the ADD NEW USER button and select an LDAP type or Active Directory type user directory in the USER DIRECTORY drop down list.

Select GROUP as the ACCOUNT TYPE.

Provide the name of the group in the ACCOUNT NAME field and click the FIND ACCOUNT button to look up the group in the LDAP or Active Directory. If the lookup succeeds then the ACCOUNT DISPLAY NAME and the DISTINGUISHED NAME fields are filled automatically.
Do provide the group name precisely. There is no functionality to look up similar names or partial matches.

The field ASSIGN TO ROLES lists all the roles that are assigned to all users of the selected group. See section Assign roles to a user, LDAP group or Active Directory group for details about role assignment.

**Assign roles to a user, LDAP group or Active Directory group**

Role(s) must be assigned to users, LDAP groups and Active Directory groups in order they can use the Web Management Console. The roles assigned to a certain user determine what pages that user can access in the Web Management Console and whether she can only read, or modify as well values on a page.

The field ASSIGN TO ROLES in the Add/assign new user(s) and Modify user dialogs lists all the roles that are assigned to the user.

The following is the role assignment policy:

1. At least one role must be assigned to a user, LDAP group or Active Directory group
2. Optionally multiple different roles can be assigned
   a. In this case the most permissive available right applies to each function.

   - For details about the permissiveness of roles see the **Roles tab** section.

**Example:**

<table>
<thead>
<tr>
<th>Roles assigned</th>
<th>Effective permissions</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Read only permission</td>
</tr>
<tr>
<td></td>
<td>Config history, Global settings</td>
</tr>
<tr>
<td>security_auditor</td>
<td></td>
</tr>
<tr>
<td>help_desk</td>
<td></td>
</tr>
<tr>
<td>security_admin AND security_auditor</td>
<td></td>
</tr>
</tbody>
</table>

|                              | Config history                        |
|                              | Global settings                        |
|                              |                                      |
Delete user

⚠ Active sessions of the deleted user will be aborted at the time of the next interaction with the server.

Special user accounts

Some user accounts are reserved in the product for system internal usage. These accounts are documented in this section.

⚠ The special accounts documented in this section are for internal usage. Do not directly modify these accounts through the user management functions cause it may give unexpected results.

SYSTEM/management account

The SYSTEM/management account is reserved for Central Management.

When the product is connected to Central Management as a managed instance, then this account is automatically created by Central Management at the first successful connection with the following parameters:

<table>
<thead>
<tr>
<th>Username</th>
<th>Password</th>
<th>Name</th>
<th>Email</th>
<th>Roles</th>
<th>User directory</th>
</tr>
</thead>
<tbody>
<tr>
<td>management</td>
<td>N/A</td>
<td>Metadefender</td>
<td>management@localhost</td>
<td>Administrators</td>
<td>SYSTEM</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Central Management</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

All consecutive connection attempts are performed by Central Management using the SYSTEM/management account.

Roles tab

Roles can be assigned to users. This simplifies controlling permissions. The Roles tab lists the existing roles in the system.
Default roles

After installation the following default roles are created with the following parameters:

<table>
<thead>
<tr>
<th>Rolename</th>
<th>Display name</th>
<th>Default member username</th>
<th>Permissions</th>
</tr>
</thead>
<tbody>
<tr>
<td>admin</td>
<td>Administrators</td>
<td>admin</td>
<td>Full on all functions</td>
</tr>
<tr>
<td>security_admin</td>
<td>Security administrators</td>
<td></td>
<td>Full on ICAP history, Security rules, Server profiles and Global settings</td>
</tr>
<tr>
<td>security_auditor</td>
<td>Security auditor</td>
<td></td>
<td>Read-only on all functions</td>
</tr>
<tr>
<td>help_desk</td>
<td>Help desk</td>
<td></td>
<td>Read-only on ICAP history, Security rules, Server profiles and Global settings</td>
</tr>
</tbody>
</table>
## Permissions

Each role has a set of rights associated to it. Each of these rights represent the level of access to the appropriate function of the MetaDefender product’s Web Management Console.

A right can be set to one of three different values:

<table>
<thead>
<tr>
<th>Right</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>NONE</td>
<td>Users with this effective right have no right to access the given function of the MetaDefender product’s Web Management Console. The menu belonging to the function is not displayed.</td>
</tr>
<tr>
<td>READ-ONLY</td>
<td>Users with this effective right are granted to access the given function for observation purposes only. Users of this role can, however, not effectuate any modifications or any change to the function.</td>
</tr>
<tr>
<td>FULL</td>
<td>Users with this effective right have full access to the given function, including viewing any data belonging to it and modifying its configuration.</td>
</tr>
</tbody>
</table>

### Effective right

A single user may have multiple roles assigned to it. There may be cases, when one of the assigned roles of the user would prohibit, while the other assigned role of the user would permit a certain function. In this case the more permissive right will be effective.

<table>
<thead>
<tr>
<th>Right</th>
<th>More permissive</th>
<th>More restrictive</th>
</tr>
</thead>
<tbody>
<tr>
<td>FULL</td>
<td></td>
<td></td>
</tr>
<tr>
<td>READ-ONLY</td>
<td></td>
<td></td>
</tr>
<tr>
<td>NONE</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

See section Assign roles to a user, LDAP group or Active Directory group for details about assigning multiple roles to a single user.

### Functions

Besides listing existing roles the Roles tab provides the following functions:

- Add new role
- Modify (and view) existing role
• Delete existing role

⚠️ The default role Administrators can not be deleted or modified.

Modify role

⚠️ The users' permissions won't be modified during the session, even if one of their roles are modified in the meantime.

For example:

1. A user is assigned to the role security_admin and has Full permissions on Config history
2. She can see Config history changes
3. During her session the Config history permissions are set to None for the security_admin role.
4. The logged in user can still select the Config history menu and can see the configuration changes there.

Then new permissions will be effective only after a logout and new login.

Delete role

⚠️ A role can not be deleted as long as it is assigned to any user.

As a consequence deleting a role can not affect active sessions of users.

User directories tab

Users can be organized into separate user directories. User directories help to enforce the following login policies:

1. Lockout after a number of consecutive failed login attempts
2. Disable logins for all users of the user directory

The Users tab lists the existing user directories in the system.
## Default user directories

After installation two default user directories are created with the following parameters:

<table>
<thead>
<tr>
<th>User directory type</th>
<th>Name</th>
<th>Number of failed logins before lockout</th>
<th>Lockout time [minutes]</th>
<th>Enable enhanced password policy</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>Local</td>
<td>LOCAL</td>
<td>3</td>
<td>5</td>
<td>False</td>
<td>This user directory can be used to add local user accounts to the system.</td>
</tr>
<tr>
<td>Local</td>
<td>SYSTEM</td>
<td>0</td>
<td>0</td>
<td>False</td>
<td>This user directory is used to add special, (e.g. machine-to-machine) user accounts to the system.</td>
</tr>
</tbody>
</table>
Three types of user directories exist in MetaDefender products:

1. **Local**
2. **LDAP**
3. **Active Directory**

**Local type user directories**

Local type user directories allow creating users that locally exist on the MetaDefender product. To protect user accounts of a local user directory against brute force password breaking attacks, the following policy settings may be applied to each local type user directory:

- **NUMBER OF FAILED LOGINS BEFORE LOCKOUT**: After this number of consecutive failed login attempts the account gets locked.
- **LOCKOUT TIME [MINUTES]**: The account remains locked for the given minutes.
  - When the lockout time elapses, the account lock gets released automatically.
  - Users with appropriate permission may release the account lock earlier using the **RELEASE LOCKOUT** button.
- **ENABLE ENHANCED PASSWORD POLICY**: If enabled, then the following policy is enforced for new passwords:

  **Enhanced password complexity policy**
  - The password must be at least 8 characters long;
  - The password must contain at least one of each
    - Upper case Latin letter character ([A-Z]),
Lower case Latin letter character ([a-z]),
Arabic numeral character ([0-9]);
The password must not contain the user name.

Active Directory type user directories
Active Directory type user directories allow users defined in an Active Directory to access the MetaDefender product.
Active Directory type user directories do not provide the possibility to define login policies; these policies may be defined in the Active Directory directly.

LDAP type user directories
LDAP type user directories allow users defined in an LDAP directory to access the MetaDefender product.
LDAP type user directories do not provide the possibility to define login policies; these policies may be defined in the LDAP directory directly.

Functions
Besides listing existing user directories the USER DIRECTORIES tab provides the following functions:
- Add new user directory
- Modify (and view) existing user directory
- Delete existing user directory
- Enable or disable existing user directory
- Unlock locked accounts

Add new Local type user directory
Click the ADD NEW USER DIRECTORY button and select Local in the USERDIRECTORY TYPE drop down list.
For explanation of the NUMBER OF FAILED LOGINS BEFORE LOCKOUT and LOCKOUT TIME [MINUTES] fields read the Local type user directories section.
Add new LDAP type or Active Directory type user directory

1. Click the ADD NEW USER DIRECTORY button and select LDAP or Active Directory in the USERDIRECTORY TYPE drop down list respectively. Learn more about the differences between LDAP and Active Directory type user directories.

2. Multiple LDAP or Active Directory servers can be configured to provide high availability of the directory services.
   a. Server preference:
      i. The configured servers are taken for a connection attempt in a top-bottom fashion.

   b. Failover conditions
      i. If the connection to a certain LDAP or AD server fails (for a reason other then authentication error) then
ii. The next server in the preference order is attempted.

```
  ________________________________
 |                               |
 |     Connect LDAP or AD server  |
 |_______________________________|
        |                           |
          | Connection error?         |
          |__________________________|
            |                           |
             | yes                       |
             |__________________________|
                  |                           |
                   | try next server            |
                   |__________________________|
                        |                           |
                        | no                        |
                        |__________________________|
                             |                           |
                              | no                        |
                              |__________________________|
                                   |                           |
                                   | yes                       |
                                   |__________________________|
                                        |                           |
                                        | yes                       |
                                        |__________________________|
                                             |                           |
                                             | yes                       |
                                             |__________________________|
                                                |                           |
                                                | no                        |
                                                |__________________________|
                                                     |                           |
                                                     | no                        |
                                                     |__________________________|
                                                          |                           |
                                                          | no                        |
                                                          |__________________________|
                                                               |                           |
                                                               | yes                       |
                                                               |__________________________|
                                                                    |                           |
                                                                    | yes                       |
                                                                    |__________________________|
                                                                                   |                           |
                                                                                   | no                        |
                                                                                   |__________________________|
                                                                                     |   c. Server properties
                                                                                     |   The following properties must be specified for each LDAP or AD server:
                                                                                     |   i. SERVER HOST: IP address or FQDN of the server
                                                                                     |   ii. SERVER PORT: The TCP port on which the directory service is listening
                                                                                     |   iii. ENCRYPTION: What kind of encryption to use when establishing the connection
                                                                                     |   1. None: No encryption, communication on a clear-text channel

As long as ENCRYPTION field is set to None there is no encryption used between the MetaDefender product and the LDAP or Active Directory server. All passwords and other information are sent clear-text over the network.

Use StartTLS or SSL as ENCRYPTION whenever possible.
2. StartTLS: The connection is established on a clear-text channel, then upgraded to TLS (TLS over LDAP)

3. SSL: The connection is established on a TLS encrypted channel (LDAP over TLS)

For StartTLS and SSL type encrypted connections the LDAP or Active Directory server must authenticate itself with a certificate that's validity can be verified by the MetaDefender product. This can happen in basically two ways:

a. Trusted certificate: the certificate's root certificate is issued by a trusted certificate authority (e.g. Comodo, Symantec, GoDaddy, GlobalSign, IdenTrust, DigiCert, StartCom, Entrust, Trustwave, Verizon, etc.) These root certificates are usually pre-installed in modern operating systems and as a result can be verified by the MetaDefender product.

b. Self-signed certificate: the certificate (or it's root certificate) is added to the system as a trusted root certificate. For details about adding a certificate as a trusted root certificate, see the manuals of the operating system of your MetaDefender product. For your convenience here are the necessary commands for Windows and some Linux distributions (the path to the self signed certificate file is `C:\Path\To\certificate.crt` or `/path/to/certificate.crt` on Windows or Linux respectively):

```
Windows

> certutil -addstore -f "ROOT" C:\Path\To\certificate.crt
```

3. Select whether to authenticate or not at the time of the bind request. Authentication at the time of the bind request is an additional security control for directory services like host based authentication at SSH. It may or may not be required by the server.
4. Selecting the ANONYMOUS BIND option will try to attempt to bind to the directory service with no username and password.

> If the ANONYMOUS BIND option is selected, then the values specified for BIND USERNAME and BIND PASSWORD are ignored, their text fields are disabled.

> Some LDAP and Active Directory servers may be configured to not permit anonymous bind requests.

b. Specify the BIND USERNAME and BIND PASSWORD values. These values must be the name as DN (distinguished name) and password of a user who has permissions to do searches in the directory.

> As long as TLS is not configured for the Web Management Console, passwords are sent clear-text over the network. For details see Transport Layer Security.

4. The USER BASE DN and the GROUP BASE DN values should provide the entries in the LDAP or Active Directory tree where user and group entity lookups should be started.

> For further LDAP specific property details see LDAP attributes.

> For further Active Directory specific property details see Active Directory attributes.

5. Click the TEST button to test the LDAP or Active Directory settings. If the test succeeds then the user directory can be added to the list with the ADD button.
Add user directory

USER DIRECTORY TYPE

Active Directory

NAME

NAME

ACTIVE DIRECTORY SETTINGS

SERVER HOST

SERVER HOST

PORT

PORT

ENCRYPTION

Delete

Add Server

ANONYMOUS BIND

BIND USERNAME

USERNAME

BIND PASSWORD

PASSWORD

USER BASE DN

USER BASE DN

GROUP BASE DN

GROUP BASE DN

ADD  CANCEL  TEST
Add user directory

USER DIRECTORY TYPE
LDAP

NAME
NAME

LDAP SETTINGS

SERVER HOST
SERVER PORT
ENCRYPTION
Delete

Add Server

ANONYMOUS BIND

BIND USERNAME
USERNAME

BIND PASSWORD
PASSWORD

USER BASE DN
USER BASE DN

GROUP BASE DN

LDAP USER SCHEMA SETTINGS

USER OBJECT CLASS
USER OBJECT CLASS

USER ACCOUNT ATTRIBUTE
USER ACCOUNT ATTRIBUTE

USER EMAIL ATTRIBUTE
USER EMAIL ATTRIBUTE
Differences between LDAP and Active Directory type user directories

From MetaDefender products' perspective LDAP directories are supersets of Active Directory directories. In other words, an Active Directory configuration can be specified using LDAP user directory type, setting certain properties to predefined values.

The following properties must be set to the following values to implement an Active Directory type user directory with LDAP type user directory
<table>
<thead>
<tr>
<th>Property</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>USER OBJECT CLASS</td>
<td>user</td>
</tr>
<tr>
<td>USER ACCOUNT ATTRIBUTE</td>
<td>samaccountname</td>
</tr>
<tr>
<td>USER EMAIL ATTRIBUTE</td>
<td>mail</td>
</tr>
<tr>
<td>USER DISPLAY NAME ATTRIBUTE</td>
<td>cn</td>
</tr>
<tr>
<td>GROUP OBJECT CLASS</td>
<td>group</td>
</tr>
<tr>
<td>GROUP ACCOUNT ATTRIBUTE</td>
<td>samaccountname</td>
</tr>
<tr>
<td>GROUP DISPLAY NAME ATTRIBUTE</td>
<td>cn</td>
</tr>
</tbody>
</table>

**Delete user directory**

⚠️ Users of the deleted user directory will be deleted as well. As a consequence: active sessions of the users of the deleted user directory will be aborted at the time of the next interaction with the server (for details see Delete user).

**Enable or disable user directory**

To disable a user directory hover over the user directory's entry in the list and click the Disable user directory icon.

When disabling a user directory, all users that are assigned to it will be blocked from logging in.

⚠️ Active sessions of users of the disabled user directory will not be aborted. The user will be blocked at the time of the next login.
When a user directory is disabled then the user directory's entry in the list displays the x mark. To enable the user directory click the *Enable user directory* icon.

**Unlock locked accounts**

All the locked user accounts that belong to a Local type user directory, can be released clicking the RELEASE LOCKOUT button.

**Transport Layer Security**

Transport Layer Security (TLS) is a cryptographic protocol that provides communications security over a computer network. Websites, like the Web Management Console, are able to use TLS to secure all communications between their servers and web browsers.

The TLS protocol aims primarily to provide confidentiality (privacy) and data integrity between two communicating computer applications.

For instructions to set up TLS see 3.2 Configuring TLS

**LDAP attributes**

For further details see 3.3.3 LDAP attributes

**Active Directory attributes**

For further details see 3.3.2 Active Directory attributes

**Notes**

The currently logged on user can not disable the user directory to which her account is assigned to. For example the *admin* user can not disable the LOCAL user directory.

The currently logged on user can not delete the following:

- Her own user account. For example the *admin* user can not delete the *admin* user account.
- The user directory to which her account is assigned to. For example the *admin* user can not delete the LOCAL user directory.

**3.3.1 Change user password**

The current user can change her password under *Settings > Password*. 
The password change option is only available for users of Local type user directories.
Changing passwords in LDAP or Active Directory services is not supported.

If enhanced password policy is enabled for the user directory this user belongs to then the new password must fulfill the password complexity requirements listed at the Local type user directories section.

**Important notes**

As long as TLS is not configured for the Web Management Console, passwords are sent clear-text over the network. To set up TLS see Configuring TLS.

**3.3.2 Active Directory attributes**

- Bind username
User base and group base DN

This page contains tips on how to obtain the BIND USERNAME, the USER BASE DN and GROUP BASE DN attributes when creating an Active Directory type user directory.

⚠️ Normally a domain administrator should provide these values.

✔️ All three attributes must be expressed using a valid LDAP syntax.
Bind username

Normally a domain administrator should provide these values, however there is a way to get the BIND USERNAME as an LDAP DN, that is needed for the product to do searches in the directory information tree, and it is as follows:
1. Log on to a Windows server machine that has connectivity to the Active Directory
2. Choose a user that is intended for this purpose (ie: has rights to do searches in the tree)
3. Open a Command window with elevated rights (Run as Administrator)
4. Assuming example.com as domain and John Smith with account name john.smith as the user, type the following:

   > dsquery user domainroot -samid john.smith

or

   > dsquery user domainroot -name John Smith

The commands above will return the correct DN for the user in question. The DN should look something like this:

   CN=John Smith,OU=People,OU=Engineering,DC=example, DC=com

Please note, the actual user DN will not look exactly like the above example, but will depend on the structure of the underlying directory information tree in the Active Directory server.

On non-server Windows machines the dsquery command can be obtained with installing the Remote Server Administration Tools (RSAT).

User base and group base DN

Once the bind user DN is obtained, an easy way to get the DNs for the user and group searches is by taking all the DC parts of the user DN and leaving the rest out, which results in the following DN:

   DC=example,DC=com
Please note that using only DC components for the user/group DNs may result in searches to be executed from the top of the directory information tree and potentially slow down AD server responses a lot and thus have an impact on the MetaDefender product's password validation. The rule of thumb here is that the more specific the user/group DN the faster the server response.

Taking the above example into consideration: a user search DN of "OU=People,OU=Engineering,DC=example,DC=com" could potentially result in much faster server response than "DC=example,DC=com" and should be preferred assuming all users reside under "OU=People,OU=Engineering,DC=example,DC=com" in the directory information tree.

Please also note that users and groups may reside in different parts of the directory information tree, as a consequence applying the same, more specific DN both as USER BASE DN and GROUP BASE DN may cause the MetaDefender product to not find group accounts in the directory information tree. So these DNs should be chosen carefully.

3.3.3 LDAP attributes

- Bind username
- User base and group base DN
- LDAP user schema settings
  - Example
- LDAP group schema settings

This page contains tips on how to obtain the BIND USERNAME, the USER BASE DN and GROUP BASE DN attributes, the LDAP USER SCHEMA SETTINGS and the LDAP GROUP SCHEMA SETTINGS when creating an LDAP type user directory.

Normally a domain administrator should provide these values.
The BIND USERNAME, the USER BASE DN and GROUP BASE DN attributes must be expressed using a valid LDAP syntax.
Add user directory

USER DIRECTORY TYPE
LDAP

NAME
NAME

LDAP SETTINGS

SERVER HOST
PORT
ENCRIPTION

SERVER HOST
PORT

BIND USERNAME
USERNAME

BIND PASSWORD
PASSWORD

USER BASE DN
USER BASE DN

GROUP BASE DN
GROUP BASE DN

LDAP USER SCHEMA SETTINGS

USER OBJECT CLASS
USER OBJECT CLASS

USER ACCOUNT ATTRIBUTE
USER ACCOUNT ATTRIBUTE

USER EMAIL ATTRIBUTE
USER EMAIL ATTRIBUTE
Bind username

Normally an LDAP administrator should provide these values, however there is a way to get the BIND USERNAME as an LDAP DN, that is needed for the product to do searches in the directory information tree. To get this information, there is, however, some information about the LDAP tree, that must be known in advance:

<table>
<thead>
<tr>
<th>LDAP property</th>
<th>Notes</th>
<th>Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>DN of the LDAP tree node that has the bind user as a</td>
<td>This will be the base for the search. If selected improperly then the</td>
<td>ou=users,dc=test</td>
</tr>
<tr>
<td>child.</td>
<td>search may be very slow or won't find the appropriate user.</td>
<td></td>
</tr>
<tr>
<td>Attributes and their values that uniquely identify the</td>
<td>These will be used as filter conditions to find the proper user.</td>
<td>uid=bind</td>
</tr>
<tr>
<td>bind user account.</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

To find the bind user, perform the following steps:

1. Log on to a Windows machine that has connectivity to the LDAP server
2. Choose a user that is intended for this purpose (i.e., has rights to do searches in the tree)

3. Open the LDP.exe tool with elevated rights (Run as Administrator)

4. Assuming the example properties above are correct, do the following search:

   ![Search Interface]

   The search above will return the attributes for the user in question:

   ![Search Results]

   The DN should look something like this:

   `cn=bind, ou=users, dc=test`
Please note, the actual user DN will most probably look completely different than the above example, as it depends on the structure of the underlying directory information tree in the LDAP server.

On non-server Windows machines the LDP.exe tool can be obtained with installing the Remote Server Administration Tools (RSAT).

**User base and group base DN**

Once the bind user DN is obtained, an easy way to get the DNs for the user and group searches is by taking all the DC parts of the user DN and leaving the rest out. Using the examples above the result DN will be the following:

```
dc=test
```

Please note that using only DC components for the user/group DNs may result in searches to be executed from the top of the directory information tree and potentially slow down the LDAP server responses a lot and thus have an impact on the MetaDefender product's password validation. The rule of thumb here is that the more specific the user/group DN, the faster the server response is.

Please also note that users and groups may reside in different parts of the directory information tree, as a consequence applying the same, more specific DN both as USER BASE DN and GROUP BASE DN may cause the MetaDefender product to not find group accounts in the directory information tree. So these DNs should be chosen carefully.

**LDAP user schema settings**

Similarly to the search in **Bind username**, we can search for users and determine the appropriate values.
Example

1. Let's assume the following search is executed:

![Search Interface](image)

2. That gives the following result:

![Search Result](image)

3. The LDAP USER SCHEMA SETTINGS may be the following:

<table>
<thead>
<tr>
<th>User schema setting</th>
<th>LDAP attribute</th>
</tr>
</thead>
<tbody>
<tr>
<td>USER OBJECT CLASS</td>
<td>user</td>
</tr>
<tr>
<td>User schema setting</td>
<td>LDAP attribute</td>
</tr>
<tr>
<td>------------------------------------------</td>
<td>----------------------</td>
</tr>
<tr>
<td>USER ACCOUNT ATTRIBUTE</td>
<td>uid or samaccountname</td>
</tr>
<tr>
<td>USER EMAIL ATTRIBUTE</td>
<td>mail</td>
</tr>
<tr>
<td>USER DISPLAY NAME ATTRIBUTE</td>
<td>cn</td>
</tr>
</tbody>
</table>

**LDAP group schema settings**

Similarly to the search in [LDAP user schema settings](#), we can search for users and determine the appropriate values.

### 3.5 Logging

MetaDefender ICAP Server has wide variety of options to configure logging. Log settings are in the configuration files or in the Windows Registry. To see more details about log configuration see the following pages:

- 3.5.1 Configuration
- 3.5.2 Debug logging
- 3.5.3 Web Management Console logs
- 3.5.4 Logging traffic of bad requests

#### 3.5.1 Configuration

**Logging configuration**

To configure the log outputs and levels, consult the [3.1.2 MetaDefender ICAP Server configuration file](#) section.

**Linux log rotation**

The installer configures the `logrotate` service to rotate MetaDefender ICAP server log files.

Configuration file is located in:

```
/etc/logrotate.d/mdicapsrv
```
The default configuration will rotate daily and store the last 30 days.

If the log file path is modified, the logrotate config file should be updated as well.

The new log settings will be used after a service restart or a HUP signal.

3.5.2 Debug logging

Debug logging can be used to identify issues with connections. It provides debug level details for each connection as well as the time each step took (e.g. parsing time, scan time, response time). It also tracks the number of active connections.

- Important notes
- Step-by-step guide
  - Windows
    - Enable debug logging
    - Disable debug logging
  - Linux (CentOS syntax)
    - Enable debug logging
    - Disable debug logging

Important notes

- Debug logging is not designed to be constantly enabled. It should only be used for investigating issues for short periods of time.
  
  Keeping it enabled permanently may impact performance. If running for too long, the log file can become huge and significantly reduce the available disk space.

- In each case below, both of logfile and loglevel must be set or none of them.
Step-by-step guide

Windows

Enable debug logging
Perform the following steps to enable debug logging:

1. Make sure MetaDefender ICAP Server is stopped
   
   ```
   > net stop mdicapsrv
   ```

2. Open the Windows Registry with regedit. Go to the
   HKEY_LOCAL_MACHINE\SOFTWARE\OPSWAT\ICAP Server\logger entry

3. Add or modify the following entries with the following values:
   a. loglevel: debug
   b. wineventlog_level: debug

4. Close the registry editor
5. Start MetaDefender ICAP server
   
   ```
   > net start mdicapsrv
   ```

Disable debug logging
Perform the following steps to disable debug logging:

1. Make sure MetaDefender ICAP Server is stopped
   
   ```
   > net stop mdicapsrv
   ```

2. Open the Windows Registry with regedit. Go to the
   HKEY_LOCAL_MACHINE\SOFTWARE\OPSWAT\ICAP Server\logger entry

3. Add or modify the following entries with the following values:
   a. loglevel: info
b. wineventlog_level: info

4. Close the registry editor
5. Start MetaDefender ICAP server

> net start mdicapsrv

**Linux (CentOS syntax)**

**Enable debug logging**

Perform the following steps to enable debug logging:

1. Make sure MetaDefender ICAP Server is stopped

    # service mdicapsrv stop

2. Edit /etc/mdicapsrv/mdicapsrv.conf

3. Add or modify the following entry under the [logger] section with the following value:

   a. loglevel: debug

4. Save and close the configuration file
5. Start MetaDefender ICAP server

    # service mdicapsrv start

**Disable debug logging**

Perform the following steps to enable debug logging:

1. Make sure MetaDefender ICAP Server is stopped

    # service mdicapsrv stop

2. Edit /etc/mdicapsrv/mdicapsrv.conf

3. Add or modify the following entry under the [logger] section with the following value:

   a. loglevel: info
4. Save and close the configuration file
5. Start MetaDefender ICAP server

```bash
# service mdicapsrv start
```

### 3.5.3 Web Management Console logs
The MetaDefender ICAP Server Web Management Console and REST interface is hosted by the nginx web server. Web Management Console and REST interface logs are written by nginx.

**Linux**

On Linux nginx logs to syslog. For details about syslog configuration see your distribution's manual.

**Windows**

Let us assume that MetaDefender ICAP Server is installed in:

```
C:\Path\To\Metadefender ICAP Server
```

In this case on Windows nginx logs to the following file:

```
C:\Path\To\Metadefender ICAP Server\nginx\nginx_error.log
```

### 3.5.4 Logging traffic of bad requests

- **Important notes**
  - **Windows**
    - Enable logging raw TCP traffic
    - Disable logging raw TCP traffic
  - **Linux (CentOS syntax)**
    - Enable logging raw TCP traffic
    - Disable logging raw TCP traffic
Logging raw TCP traffic can be used to identify issues with bad requests. It provides debug level details for requests that were refused by ICAP Server's request parser due to syntax errors.

**Important notes**

- Logging raw TCP traffic is not designed to be constantly enabled. It should only be used for investigating issues for short periods of time. Keeping it enabled permanently may impact performance. If running for too long, the log database can become huge and significantly reduce the available disk space.

- Raw TCP traffic logs may contain sensitive or private information in a clear-text format.

**Step-by-step guide**

- For configuration details see [3.1.2 MetaDefender ICAP Server configuration file](#).

**Windows**

**Enable logging raw TCP traffic**

Perform the following steps to enable traffic logging:

1. Make sure MetaDefender ICAP Server is stopped

   ```
   > net stop mdicapsrv
   ```

2. Open the Windows Registry with `regedit`. Go to the `HKEY_LOCAL_MACHINE\SOFTWARE\OPSWAT\ICAP Server\logger` entry

3. Add or modify the following entry with the following value:

   a. `capture_traffic: 1`

4. Close the registry editor

5. Start MetaDefender ICAP server
Disable logging raw TCP traffic

Perform the following steps to disable traffic logging:

1. Make sure MetaDefender ICAP Server is stopped

   > net stop mdicapsrv

2. Open the Windows Registry with regedit. Go to the
   HKEY_LOCAL_MACHINE\SOFTWARE\OPSWAT\ICAP Server\logger entry

3. Delete or modify the following entries with the following values:
   a. capture_traffic: 0

4. Close the registry editor

5. Start MetaDefender ICAP server

   > net start mdicapsrv

Linux (CentOS syntax)

Enable logging raw TCP traffic

Perform the following steps to enable traffic logging:

1. Make sure MetaDefender ICAP Server is stopped

   # service mdicapsrv stop

2. Edit /etc/mdicapsrv/mdicapsrv.conf

3. Add or modify the following entry under the [logger] section with the following value:
   a. capture_traffic: 1

4. Save and close the configuration file

5. Start MetaDefender ICAP server
Disable logging raw TCP traffic

Perform the following steps to enable traffic logging:

1. Make sure MetaDefender ICAP Server is stopped

```
# service mdicapsrv stop
```

2. Edit `/etc/mdicapsrv/mdicapsrv.conf`

3. Delete or modify the following entry under the `[logger]` section with the following value:

```
a. capture_traffic: 0
```

4. Save and close the configuration file

5. Start MetaDefender ICAP server

```
# service mdicapsrv start
```

3.6. Internet connection

MetaDefender ICAP Server requires internet connection for online license activation.

Proxy configuration

**Linux**

Set variable `https_proxy` in file `/etc/default/mdicapsrv`.

**Windows**

Under Windows use the `netsh` tool to set the proxy, e.g.: `netsh winhttp set proxy <ADDRESS>

In some cases setting the proxy with `netsh` is not sufficient. In that case set the proxy by starting Internet Explorer with SYSTEM rights and configure the proxy in the settings. To do this please follow this article.
You might need to configure Windows proxy to bypass local addresses if you can’t access Web Management Console from the host itself. Consult netsh documentation for additional configuration options.

### 3.7 Other settings

**Global settings**

Global settings may be configured under Settings > Global settings.

The following properties are available for configuration:

1. **USE PERSISTENT CONNECTIONS**: When this option is enabled the MetaDefender ICAP Server keeps the connections open, so that they can be reused for several requests.

2. **ENABLE PERMISSIVE PARSING**: by default MetaDefender ICAP Server verifies requests strictly against the syntax specified in RFCs. If this option is enabled, then the parser is more permissive: e.g. space characters will be allowed in request URLs.
Data retention

Data retention settings may be configured under **Settings > Data retention**.

The following properties are available for configuration:

1. ICAP history clean up: ICAP history records under **Dashboard > ICAP history** will be retained for the time configured here.
   a. OFF: Switch off automatic clean up.
   b. 1 HOUR .. 12 MONTHS: ICAP history records are automatically deleted after reaching the age specified by the configuration value.

2. Config history clean up: audit records under **Dashboard > Config history** will be retained for the time configured here.
   a. OFF: Switch off automatic clean up.
   b. 1 HOUR .. 12 MONTHS: audit records are automatically deleted after reaching the age specified by the configuration value.
3.8 Customizing the block page

- Replacing the default block page
  - ICAP Server block messages
  - Additional placeholders
    - Example
  - Default block page source code

When a request is blocked for some reason then a block page is displayed that –by default– looks as follows (the actual blocking reason message may differ):

![File blocked]

This page can be completely replaced using a custom block page.

**Replacing the default block page**

To replace the default block page create a HTML file with the custom content according to the following:
<table>
<thead>
<tr>
<th>Operating system</th>
<th>Path</th>
<th>Default path</th>
<th>Owner</th>
<th>Permissions</th>
</tr>
</thead>
<tbody>
<tr>
<td>Windows or Windows Server</td>
<td><code>&lt;Installation folder&gt;\data\block_page.html</code></td>
<td><code>C:\Program Files\Metadefender ICAP Server\data\block_page.html</code></td>
<td>Local Service</td>
<td>Read</td>
</tr>
<tr>
<td>Linux (RHEL / CentOS / Debian / Ubuntu)</td>
<td><code>&lt;Data folder&gt;block_page.html</code></td>
<td><code>/var/lib/mdicapsrv/block_page.html</code></td>
<td>mdicapsrv: mdicapsrv</td>
<td>r-- (400)</td>
</tr>
</tbody>
</table>

⚠️ The default block page is not represented as a file in the file system. If the above file exists, then the default block page is overridden.

### ICAP Server block messages

To display ICAP Server block messages in the custom page include the following token at the desired position in the page:

`%%%icap_block_message%%%

This token will be replaced by the appropriate ICAP Server block message.

### Additional placeholders

For custom block pages several additional placeholder tokens are also supported.

⚠️ **Not in the default block page**

The placeholder tokens below are not used in the default block page, but are available for a custom block page.
<table>
<thead>
<tr>
<th>Placeholder</th>
<th>Description</th>
<th>Note</th>
</tr>
</thead>
<tbody>
<tr>
<td>%%%url%%%</td>
<td>URL parsed from the original HTTP request.</td>
<td></td>
</tr>
<tr>
<td>%%%client_ip%%%</td>
<td>Address of the client from whom the request originated.</td>
<td></td>
</tr>
<tr>
<td>%%%server_name%%%</td>
<td>Name of the server the ICAP Server is running on.</td>
<td></td>
</tr>
<tr>
<td>%%%timestamp%%%</td>
<td>Arrival of the request to the ICAP Server.</td>
<td></td>
</tr>
<tr>
<td>%%%filenames%%%</td>
<td>Filenames which are blocked.</td>
<td>Present only if some violations were found, otherwise replaced by empty string.</td>
</tr>
<tr>
<td>%%%mimes%%%</td>
<td>Media types for blocked files.</td>
<td></td>
</tr>
<tr>
<td>%%%violations%%%</td>
<td>Violations for blocked files.</td>
<td></td>
</tr>
<tr>
<td>%%%violations_short%%%</td>
<td>The first violation for the blocked file.</td>
<td></td>
</tr>
</tbody>
</table>

**Example**

The following example is the source code of a simple custom page that displays the ICAP Server messages only with no formatting at all:

```html
<!doctype html>
<html>
<body>
  %%%icap_block_message%%%
</body>
</html>
```

It renders a very simple page that looks like this:
End-user client’s license identifier is missing

**Default block page source code**

The default lock page's source code can be reused to create a customized one.

```html
<!doctype html>
<html lang="en">
<head>
  <title>MetaDefender ICAP server - File blocked</title>
  <link href="https://fonts.googleapis.com/css?family=Roboto:400,500" rel="stylesheet">
  <style>
    html, body { margin: 0; font-size: 14px; position: relative; height: 100%; background-color: #F9F9F8; color: #273238; font-family: 'Roboto', sans-serif; }
    a { color: inherit; }
    a:hover { color: inherit; }
    hr { margin: 0; border: 0; border-top: 1px solid #EFEFED;
```
3.9 Server profiles

Server profiles under Inventory > Server profiles help to organize services of one or more servers based on the service type.

For example in case of security rules one or more MetaDefender Core servers are needed to scan requests. For this purpose a server inventory may be created collecting all available Core servers. Then at the rule itself simply this server profile needs to be selected.
Currently MetaDefender ICAP Server uses and allows *MetaDefender Core* type server profiles only.

### Properties

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Server profile type</strong></td>
<td>Service type</td>
</tr>
<tr>
<td></td>
<td>Supported service types are:</td>
</tr>
<tr>
<td></td>
<td>1. <em>MetaDefender Core</em></td>
</tr>
<tr>
<td><strong>Profile name</strong></td>
<td>Unique identifier of the server profile</td>
</tr>
<tr>
<td><strong>Server specifications (URI)</strong></td>
<td>Service specifications in <a href="https://example.com">URI syntax</a>. Multiple server specification may be added to a server profile. At least one server specification must exist in a server profile. Only the following URI components are used: 1. <em>scheme</em> 2. <em>host</em> 3. <em>port</em> Example: <code>http://127.0.0.1:8008</code></td>
</tr>
</tbody>
</table>

⚠️ In case of *MetaDefender Core* server profile types the very same URI will be used for the URLs of *View scan details* links on the *Request details* page under **Dashboard > ICAP history**. If the URI specified here is not reachable on the machine where the actual browsing of the Web Management Console happens (e.g. it is `127.0.0.1` and browsing happens on another machine) then the *View scan details* link will be broken. See also the *Request details* section in **4.1. Dashboard**.
Property | Description
--- | ---
Possible values: 1. **FAILOVER**: high availability order; first successfully addressed server in the list will do the service 2. **ROUND ROBIN**: load balancing order; next successfully addressed server in the list will do the service

**Certificate based client authentication**
If the destination server requires certificate based client host authentication then this checkbox must be marked. MetaDefender ICAP Server will use the actual deployment's digital ID, for details see **3.2 Configuring TLS**

**Property validation**
Some of the server profile properties have cross-dependencies and as so must match.

<table>
<thead>
<tr>
<th>Server profile type</th>
<th>Server specifications (URI) allowed schemes</th>
</tr>
</thead>
<tbody>
<tr>
<td>MetaDefender Core</td>
<td>http</td>
</tr>
<tr>
<td></td>
<td>https</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Server specifications (URI) scheme</th>
<th>Transport level encryption allowed values</th>
</tr>
</thead>
<tbody>
<tr>
<td>http</td>
<td>N/A</td>
</tr>
<tr>
<td>https</td>
<td>N/A</td>
</tr>
</tbody>
</table>

> If https scheme is specified then the HTTP connection is established over TLS.
Testing the configuration

Clicking the TEST button will test the configuration. The test consists of two steps:

1. Syntactical validation of the values
2. Connection test

⚠️ If the test fails, then the server profile can not be added.

Syntactical validation

The correctness of the provided values is validated:

1. PROFILE NAME must be unique
2. The SERVER SPECIFICATIONS (URI) values must conform with the URI syntax with the restriction that only the scheme, host and port values are allowed
3. Cross dependencies must match (see the Property validation section)

Connection test

If the syntactical validation pass, then each server specification is tested for a successful connection.

Limitations

⚠️ Currently the connection is tested without using TLS (when configured at all).

3.9.1 MetaDefender Core specific inventory properties

Query parameters

Addresses specified in CORE URI ADDRESS may contain optional query parameters for specific use cases.

Syntax

With the optional query parameters the Core URI is formatted as follows:
**URI schema with query parameter**

`scheme://host:port[?query]`

### Valid parameters

<table>
<thead>
<tr>
<th>Name</th>
<th>Value</th>
<th>Use case</th>
<th>Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>apikey</td>
<td>32 characters long API key of an account that belongs to the role the Core side rule is restricted to.</td>
<td>Core side rules may be restricted to roles (for details see <a href="#">3.6.4. Workflow rule configuration</a> in MetaDefender Core v4 documentation). Email Security can use these rules if:</td>
<td><strong>URI example with query parameter</strong>&lt;br&gt;<a href="https://127.0.0.1:8058?apikey=24a586f3023b4309b65a1974198d6c6">https://127.0.0.1:8058?apikey=24a586f3023b4309b65a1974198d6c6</a></td>
</tr>
</tbody>
</table>

1. A user is created on Core,  
   a. That belongs to the role the Core side rule is restricted to;  
   b. And the user has an API key generated.  
2. In Email Security the Core is added to a server profile  
   a. And the API key above is specified in the Core URI.

### Rule

Right after the CORE URI ADDRESS has been specified, and is valid (connection can be established) the RULE drop-down list is populated with the available rules on the corresponding MetaDefender Core instance.
The desired security rule may be selected from the configured Core's available rules. Default is *Automatic*.

⚠️ Please note that if the security rules on the configured MetaDefender Core under **Policy > Security rules** are limited to a specific user agent other than `mdicapservlet` or restricted to specific roles other than what is specified in the server URI, then those security rules will not be available in MetaDefender ICAP Server.

MetaDefender ICAP Server identifies itself on MetaDefender Core with the user agent `mdicapservlet`. 
On the example below the Core rule *File process with Kiosk* and *File process with SFT* are not listed in ICAP Server as those rules are restricted to user agent *kiosk* and *sft* respectively.
Otherwise all unrestricted core rules are in the list.
4. Operating MetaDefender ICAP Server

- 4.1. Dashboard
- 4.2 Security rules
- 4.3 ICAP response headers
- 4.4 Web Proxy Integrations
- 4.5 Other Integration
- 4.6 HTTP Header Handling

4.1. Dashboard

MetaDefender ICAP Server's Dashboard gives overview of ICAP Server status.

Note that the default refresh rate of displayed information is 1 minute.
Overview page

The Overview page shows information on

- Number of blocked requests
- Number of processed requests
- License information
- Number of processed requests chart

Clicking the Settings icon the following properties can be customized:

- Refresh rate (default: 1 minute)
- Time-span displayed (default: 24 hours)

ICAP history

The ICAP history shows information on ICAP events in the system.

On the ICAP history page you can search for ACTION, RESULT, HTTP URI, CLIENT IP, PROXY IP, HOST, CORE RULE or ICAP RULE. Also you can filter the list by ACTION.

⚠️ Due to usability reasons the ICAP history list is not updated automatically. Click the Refresh icon to update.

ℹ️ The RESULT value Undisclosed means that MetaDefender Core did not return a scan result for this entry.

The reason for this value appearing in ICAP History is that MetaDefender Core is configured to provide OVERALL RESULTS ONLY for VISIBILITY OF SCAN RESULTS in the security rule that matches ICAP file scans.

Request details

Clicking an ICAP history entry displays public details (that do not require authentication on MetaDefender Core) about the scan.

The View scan details link points to the scan details on the MetaDefender Core instance where the actual scanning took place.

⚠️ The View scan details link works with MetaDefender Core version 4.7.0 (or later) only.
For MetaDefender Core versions earlier than 4.7.0 the Dashboard is opened.

To generate the View scan details links, MetaDefender ICAP Server stores the actual Core IP where the current request's files were scanned. As a result the View scan details links continue to work properly even after a new Core is configured instead the old one.

View scan details links stop working after the referenced Core is uninstalled or migrated to a new address.

Export to CSV
Clicking the EXPORT TO CSV button will export the history list (according to the actual filter conditions) to a CSV file.

Data range
The currently active filter conditions apply to the exported list. All filtered data gets exported, even if the list expands to multiple pages.

CSV file download
The CSV file is written into a HTTP stream, so it gets downloaded by the browser immediately, automatically. The CSV file is named according to the following scheme:

Naming convention
icaphistory-<yyy>-<MM>-<dd>-<HH>-<mm>-<ss>.csv

For example, if the export happened on 22nd January, 2020; at 12:38:36 PM:

Naming example
icaphistory-2020-01-22-12-38-36.csv
Exported data

The following data is exported for each entry:

- Date
- Client Address
- Proxy Address
- Duration
- ICAP Method
- ICAP Service
- ICAP Rule
- HTTP URI
- Action
- Verdict
- IsTag
- Data ID
- Host
- Core Address
- Core Rule
- Sanitized
- Scan Duration
- Error Details

The history above exports to the following CSV:
The Config history shows information on configuration changes made via the Web Management Console.

Changes made directly in the configuration file are not reflected here.

On the Config history page you can search for USER, TYPE, CHANGE TYPE, PARAMETER, OLD VALUE or NEW VALUE. Also you can filter the list by DATE.

Due to usability reasons the Config history list is not updated automatically. Click the Refresh icon to update.
Filtering columns

By clicking on the filtering columns icon (▼) a new row will appear below the table header with filters for the given columns.

Using these filters you can search for ICAP requests which have to meet multiple criteria e.g. request where the action was BLOCKED and the client IP was 127.0.0.1 at the same time. This indicates that there is AND relation between the filters.
**Tokenization**

When using text filters, like Result you would expect that writing "Ba" in the filter would return every rows with "Bad request" result however this is not the case since the search is based on tokens. If you would like to get the requests with result "Bad request" you would have to write "Ba*" or "Bad" as the filter string.

**Value only search**

By default it is possible to search for a certain value among the values of all columns.

**Example**

<table>
<thead>
<tr>
<th>List</th>
</tr>
</thead>
<tbody>
<tr>
<td>ID</td>
</tr>
<tr>
<td>1</td>
</tr>
<tr>
<td>2</td>
</tr>
</tbody>
</table>

**Search results**

<table>
<thead>
<tr>
<th>Search term</th>
<th>Matching rows</th>
</tr>
</thead>
<tbody>
<tr>
<td>BLOCKED</td>
<td>1, 2</td>
</tr>
</tbody>
</table>

**Lists with value only search**

The following lists in ICAP Server implement value only search:

1. Dashboard > Config history

**Full text search (FTS)**

MetaDefender ICAP Server implements the SQLite MATCH feature of SQLite FTS3 and FTS4 Extensions (see section 3. Full-text Index Queries). In addition to the capabilities the value only search provides, using the MATCH feature makes it possible to search for certain values in certain fields.
Example

List

<table>
<thead>
<tr>
<th>ID</th>
<th>RESULT</th>
<th>STATUS</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>ALLOWED</td>
<td>BLOCKED</td>
</tr>
<tr>
<td>2</td>
<td>BLOCKED</td>
<td>COMPLETED</td>
</tr>
<tr>
<td>3</td>
<td>NOT BLOCKED</td>
<td>COMPLETED</td>
</tr>
<tr>
<td>4</td>
<td>BLOWN</td>
<td>BLOCKED</td>
</tr>
</tbody>
</table>

Search results

<table>
<thead>
<tr>
<th>Search term</th>
<th>Matching rows</th>
</tr>
</thead>
<tbody>
<tr>
<td>BLOCKED</td>
<td>1, 2</td>
</tr>
<tr>
<td>RESULT:BLOCKED</td>
<td>1, 3</td>
</tr>
<tr>
<td>RESULT:^BLOCKED</td>
<td>1</td>
</tr>
<tr>
<td>RESULT:BLO*</td>
<td>1, 3, 4</td>
</tr>
<tr>
<td>RESULT:^BLO*</td>
<td>1, 4</td>
</tr>
</tbody>
</table>

For a more complete reference of possible search terms see [https://www.sqlite.org/fts3.html](https://www.sqlite.org/fts3.html).

Lists with FTS

The following lists in ICAP Server implement full text search:

1. Dashboard > ICAP history
4.2 Security rules

MetaDefender ICAP Server security rules can be configured under Policy > Security rules.

- Security rules overview
  - Request filter
    - Filter parts
      - Host IP/Domain conditions
      - Client IP conditions
      - Wildcard matching examples
      - ICAP client IP conditions
      - Generic conditions
  - Header references
  - Security policy
    - Scan settings
    - Data trickling
    - Advanced settings
  - Rule order
    - First match policy
    - No match policy
  - Security Rule Management
    - Default security rule
    - Functions
  - Use case examples
    - Enable anti-malware test site for test clients

Security rules overview

Security rules help to assign security policies to specific requests.

A security rule consists of two parts:

- Request filter
- Security policy definition
Request filter

Based on header values or source address, request filters select requests on which the assigned security policy will be applied. Request filters can be set up on the Request filters tab while creating or editing a security rule.

Filtering can be based on source address, ICAP, HTTP or custom headers. The following parameters must be provided to create a filter condition:

- Header name: name of the header to look up.
- ICAP client address: address of the ICAP client
- Match type: the value of the referred header must be equal to, contain, start with, or end with the provided value.
- Value: the value to match.

⚠️ If a header with the referred header name does not exist in the request, then this condition won't match.

Filter parts
A filter may consist of four parts:

1. Host IP/Domain conditions
2. Client IP conditions
3. ICAP client address
4. Generic conditions

⚠️ There is **AND** relation among the above parts of the filter.

### Host IP/Domain conditions

<table>
<thead>
<tr>
<th>Header name</th>
<th>Match type</th>
</tr>
</thead>
<tbody>
<tr>
<td>Host</td>
<td>Wildcard matching (glob)</td>
</tr>
</tbody>
</table>

The Host header usually contains the requested URI.

⚠️ There is **OR** relation among entries in Host IP/Domain conditions.

### Client IP conditions

<table>
<thead>
<tr>
<th>Header name</th>
<th>Match type</th>
</tr>
</thead>
<tbody>
<tr>
<td>X-Client-IP</td>
<td>Wildcard matching (glob)</td>
</tr>
</tbody>
</table>

The X-Client-IP header usually contains the IP of the machine from which the request was originally sent.

⚠️ There is **OR** relation among the Client IP conditions.
Wildcard matching examples

<table>
<thead>
<tr>
<th>Example pattern</th>
<th>Matches</th>
<th>Does not match</th>
</tr>
</thead>
<tbody>
<tr>
<td>127.0.0.*</td>
<td>127.0.0.1, 127.0.0.150</td>
<td>127.0.100.1, 127.5.0.1</td>
</tr>
<tr>
<td><a href="http://www.example.co">www.example.co</a>?</td>
<td><a href="http://www.example.com">www.example.com</a></td>
<td><a href="http://www.example.co">www.example.co</a></td>
</tr>
<tr>
<td>[ab]cd</td>
<td>acd, bcd</td>
<td>cd, ecd</td>
</tr>
<tr>
<td>abc</td>
<td>abc</td>
<td>cba, xyz</td>
</tr>
</tbody>
</table>

Please note the difference between the Client IP conditions and the ICAP client IP conditions: the client IP is the node from whom the original request is sent, while the ICAP client is the network element that is in direct connection with the ICAP Server (e.g. proxy).

ICAP client IP conditions

The ICAP client address is the address from where the ICAP request is originated (for example a proxy server).

**Match type**

The match type is wildcard matching (glob).

There is OR relation among the ICAP client address conditions.

Please note the difference between the Client IP conditions and the ICAP client IP conditions: the client IP is the node from whom the original request is sent, while the ICAP client is the network element that is in direct connection with the ICAP Server (e.g. proxy).
Generic conditions

Using generic conditions, filters can be set up to any headers.

As an example Host IP/Domain and Client IP conditions could also be set up as generic conditions.

There is AND relation among the generic conditions.

Header references

Filtering can be based on ICAP, HTTP or even custom headers. For an incomplete reference of possible headers consult the following sources:

- HTTP
  - IANA
  - Wikipedia
- ICAP: IETF
• **Custom:** Initially, it was recommended to begin naming custom headers with X-, however, according to [RFC 6648](https://tools.ietf.org/html/rfc6648), this recommendation has since been deprecated.

**Security policy**

A security policy consists of the SCAN, DATA TRICKLING and the ADVANCED settings.

**Scan settings**

The scan policy can be set up on the Scan tab while creating or editing a security rule.

The following options are available:

1. **ALLOW SCAN:** enable or disable scanning of requests. Default is enabled.
   a. **METADEFENDER CORE:** MetaDefender Core type server profile to which requests are sent for scanning. Default is **None**.
   b. **SCAN TIMEOUT:** ICAP Server will try to get the scan results for the timeout defined here. If the timeout elapses then the request gets blocked. Default is 0.
      i. Valid range: 0 .. 86400
      ii. 0 (zero) means that there is no timeout: ICAP Server will wait for the results for any long time.

   > When scanning is disabled, all requests will be accepted.

**Data trickling**

Data trickling preferences can be set on the Data trickling tab while creating or editing a security rule. For further details about data trickling see [4.7 Data trickling](#).

The following options are available:

1. **ENABLE DATA TRICKLING:** enable releasing drips of the original contents even before the scanning completes.

   > **Risk of potential threats allowed**

   **WARNING!**

   Given by the nature of the thing, enabling data trickling may cause the ICAP Server to release potentially malicious content.

   Always enable data trickling with care; only if you know what you are doing, if you are aware of the risk, and if you accept this risk.
a. FIRST DRIP: settings related to the first drip of the original content
   i. DELAY: Delay of the first drip –after the contents have been submitted for scanning– in seconds.
   ii. SIZE: Size of the first drip –after the contents have been submitted for scanning– in kilobytes.

b. ADDITIONAL DRIPS
   i. DELAY: Delay of each consecutive drips –after the first, or any other previous drip– in seconds.
   ii. SIZE: Size of each consecutive drips –after the first, or any other previous drip– in kilobytes.

c. ENABLE TO WITHHOLD: Whether to allow or not to withhold the last portion of the original contents from being dripped before the scanning completes.

⚠️ Risk of releasing malicious object

If ENABLE TO WITHHOLD is not set, and the end of the file is reached any time during the trickling, then the whole original, potentially malicious content has been already released.

Always set ENABLE TO WITHHOLD, except only if you know what you are doing, if you are aware of the risk, and if you accept this risk.

i. Size: Size of withheld data in kilobytes.

Advanced settings

Some default security behaviors of the system can be modified on the Advanced tab while creating or editing a security rule.

The following options are available:

1. OVERRIDE SCAN RESULTS CLASSIFIED AS ALLOWED: enable or disable overriding the following default security behaviors all together.

   a. MULTIPART PARSING ERROR: Requests with multipart content are parsed for boundary and control header correctness. By default requests with multipart parsing error are rejected by MetaDefender ICAP Server. Enabling this option will accept such requests. (For further details about multipart content see RFC 1341 and RFC 2388.)
b. **CORE BUSY:** By default MetaDefender ICAP Server rejects requests when none of the Cores of the applied server profile can accept the scan requests due to overload. Enabling this option scanning will be skipped and the request will be accepted when all Cores are busy.

c. **CORE RULE FILE SIZE LIMIT EXCEEDED:** By default MetaDefender Core blocks a file that exceeds the file size limit configured for the Core rule (see 3.6.2. Workflow template configuration and 3.6.4. Workflow rule configuration chapters in MetaDefender Core v4 User Guide) applied for the ICAP Server security rule. Enabling this option results the oversized request to bypass scanning.

d. **SCAN TIMEOUT:** By enabling this option MetaDefender ICAP Server will allow requests whose scan timed out.

e. **CORE SERVER ERROR:** By enabling this option MetaDefender ICAP Server will allow requests which had a general MetaDefender Core error during their processing. This means every error where MetaDefender Core responded with the status code 500 and is not listed above as a specific case.

2. **ACTION FOR NOT SUPPORTED ENCODINGS:** MetaDefender ICAP Server supports gzip, deflate, brotli and base64 encoding only. By default requests with any other encoding are rejected by default. Here you can select your preferred action for these kind of requests.

   a. **BLOCK:** Block the request. This is the default option.

   b. **SCAN WITHOUT DECODING:** Send the content without decoding towards MetaDefender Core for processing and continue serving the request based on the processing result.

   c. **ALLOW:** Allow the request without processing the content.

3. **ACTION FOR DECODING ERRORS:** MetaDefender ICAP Server supports gzip, deflate, brotli and base64 encoding of request contents. By default requests with encoding error are rejected. Here you can select your preferred action for these kind of requests.

   a. **BLOCK:** Block the request. This is the default option.

   b. **SCAN WITHOUT DECODING:** Send the content without decoding towards MetaDefender Core for processing and continue serving the request based on the processing result.

   c. **ALLOW:** Allow the request without processing the content.
Rule order

Several security rules can be created that may target different requests from different hosts going to different URL. However, care must be taken how these rules are set up and ordered, as there is a first match and a no match policy.

Specific rules should come first while generic rules should go at last.

First match policy

If there are more matching rules in the system, then the request will be accepted or rejected according to the security policy of the first matching rule in the list.

No match policy

If there is no matching rule in the system (or no rule at all), then the request will be rejected.

The goal of the 147167977 is to have a rule in the system that matches in the end.

Deleting all security rules from the system results all requests being blocked.

Security Rule Management

The Security Rule Management screen lists the existing security rules in the system.
Default security rule

After installation a default security rule is created with the following parameters:

<table>
<thead>
<tr>
<th>Name</th>
<th>Request filters</th>
<th>Allow scan</th>
<th>Applied core rule</th>
<th>Advanced</th>
</tr>
</thead>
<tbody>
<tr>
<td>Default</td>
<td>none</td>
<td>Yes</td>
<td>Automatic</td>
<td>none</td>
</tr>
</tbody>
</table>

⚠️ The Default security rule matches any request. It will enforce scanning with automatically selected Core rule. It does not override any default advanced setting.

Functions

Besides listing existing security rules the Security Rule Management screen provides the following functions:

- Add new security rule
- Clone an existing security rule
• Modify (and view) existing security rule's properties
• Delete existing security rules

Use case examples

Enable anti-malware test site for test clients
The site www.eicar.org is created to test anti-malware solutions. Under the URI http://www.eicar.org/download/eicar.com.txt there is a malware test file that is forbidden by MetaDefender ICAP Server by default:

```bash
$ wget http://www.eicar.org/download/eicar.com.txt
Connecting to 127.0.0.1:3128... connected.
Proxy request sent, awaiting response... 403 Forbidden
2017-04-18 17:06:03 ERROR 403: Forbidden.
```

Create the following rule (in the actual test the client browser, the proxy and the MetaDefender ICAP Server is on the same machine, that is the reason for the Client IP setting 127.0.0.1):
Place it before the *Default* rule. Now the request is served:

```
$ wget http://www.eicar.org/download/eicar.com.txt
Connecting to 127.0.0.1:3128... connected.
Proxy request sent, awaiting response... 200 OK
Length: 68 [application/octet-stream]
Saving to: `eicar.com.txt.3`

100%[================================================================] 68          --.-K
/s   in 0s

2017-04-18 17:15:56 (20.2 MB/s) - `eicar.com.txt.3` saved [68/68]
```
### 4.3 ICAP response headers

Besides the standard ICAP headers following response headers are used by the MetaDefender ICAP Server:

<table>
<thead>
<tr>
<th>Header name</th>
<th>Description</th>
<th>Example</th>
<th>Note</th>
</tr>
</thead>
<tbody>
<tr>
<td>X-Blocked-Reason</td>
<td>MetaDefender specific custom header. Contains the blocking reason of the content.</td>
<td>X-Blocked-Reason: Infected</td>
<td>It is available only if the content was scanned and some violations were found.</td>
</tr>
<tr>
<td>X-ICAP-Profile</td>
<td>Contains the applied workflow's name.</td>
<td>X-ICAP-Profile: Proxy</td>
<td>It is available only if the file was scanned.</td>
</tr>
<tr>
<td>X-Response-Info</td>
<td>Contains the one word description of the action the ICAP server applied on the request.</td>
<td>X-Response-Info: Allowed</td>
<td>This header is available in all responses sent by the ICAP server.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>X-Response-Info: Blocked</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>X-Response-Info: Options</td>
<td></td>
</tr>
<tr>
<td>X-Response-Desc</td>
<td>Contains the blocking reason.</td>
<td>X-Response-Desc: Infected</td>
<td>The header is available in all &quot;blocked&quot; responses.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>X-Response-Desc: Encrypted Archive</td>
<td>In case of the content was scanned and some violations were found, the returned string is equivalent to X-Blocked-Reason's value.</td>
</tr>
<tr>
<td>X-Virus-ID</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Header name</td>
<td>Description</td>
<td>Example</td>
<td>Note</td>
</tr>
<tr>
<td>----------------------</td>
<td>------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
<td>---------------------------------------------------------------------------------------------------------------------------</td>
<td>-------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
</tbody>
</table>
|                      | Contains a short description of the threat that was found in the content. If multiple threats were found, only the first one is returned.                                                                 | X-Virus-ID: EICAR Test String  
X-Virus-ID: Encrypted Archive                                                                 | The header is available only if the content was scanned and some violations were found.                                   |
| X-Infection-Found    | Contains the description of the threat that was found in the content. If multiple threats were found, only the first one is returned.  
The value is a semicolon separated list with three parameters:  
- Type  
  - 0: Infection has been found  
  - 2: Container violation has been found  
- Resolution:  
  - 0: The suspicious content was not repaired  
- Threat: Threat name | X-Infection-Found:  
Type=0; Resolution=0;  
Threat=EICAR Test String;  
X-Infection-Found:  
Type=2; Resolution=0;  
Threat=Encrypted Archive; | The header is present only if the content was scanned and some violations were found.                                                                 |
| X-Violations-Found   | Contains the detailed description of the violations that were found. If the scanned content was an archive, the scan results for the contained files too are                                                                 | X-Violations-Found:  
test.zip  
EICAR Test String  
0  
0  
\eicar.txt                                                                 | The header is present only if the content was scanned and some violations were found.                                    |
<table>
<thead>
<tr>
<th>Header Name</th>
<th>Description</th>
<th>Example</th>
<th>Note</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>listed. If multiple threats were found for a single file, only the first one is returned. The structure of the header value is the following: The first line contains the number of the reported violations. The following lines contain the details. Filename Threat name ProblemID (currently 0 returned for all threats) ResolutionID:</td>
<td>EICAR Test String 0 0</td>
<td></td>
</tr>
<tr>
<td>X-Include</td>
<td>Contains the list of requested headers, that the ICAP clients should add to the requests, if the information is available.</td>
<td>X-Include: X-Client-IP</td>
<td>The header is present only in Options responses.</td>
</tr>
</tbody>
</table>

### 4.4 Web Proxy Integrations

MetaDefender Core can be integrated with any client that implements the standard ICAP interface, via the MetaDefender ICAP Server.

MetaDefender ICAP Server is *compatible* with any client that implements the standard ICAP interface. Moreover, MetaDefender ICAP Server *supports* ICAP clients, that were tested by the development team. The supported ICAP clients are listed in section Supported ICAP Clients below.
ICAP URL

<table>
<thead>
<tr>
<th>Mode</th>
<th>URL</th>
<th>Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>Request Modification Mode (REQMOD, OPTIONS)</td>
<td>icap://&lt;IP address&gt;:&lt;port&gt;/OMSScanReq-AV</td>
<td>e.g, icap://10.0.50.36:1344/OMSScanReq-AV</td>
</tr>
<tr>
<td>Response Modification Mode (RESPMOD, OPTIONS)</td>
<td>icap://&lt;IP address&gt;:&lt;port&gt;/OMSScanResp-AV</td>
<td>e.g, icap://10.0.50.36:1344/OMSScanResp-AV</td>
</tr>
</tbody>
</table>

Supported ICAP Clients

- 4.4.1 ARA network JAGUAR5000
- 4.4.2 Symantec Blue Coat Proxy SG
- 4.4.3 F5 Integrations
- 4.4.4 McAfee Web Gateway
- 4.4.5 Squid
- 4.4.6 FortiGate
- 4.4.7 A10 SSL Insight (SSLi)

4.4.1 ARA network JAGUAR5000

Configuration via Web-based manager

1. In your browser log into Jaguar's Web Administration
2. Go to the Configuration page
3. From the menu on the left, click on Others under the Protocol group
4. From the top menu, select ICAP
5. Make sure that Send Client Address checkbox is checked
6. Apply changes
7. Click on the **ICAP Profile** button on the bottom of the page
8. Add a new profile
9. Type in "icap://<IP>:<PORT>/OMSScanReq-AV" to **Vectoring Point 1**
10. Type in "icap://<IP>:<PORT>/OMSScanResp-AV " to **Vectoring Point 3**
11. Apply changes

---

**ICAP Profile**

---

**Scanning HTTPS Content**

1. Go to the Configuration page
2. From the menu on the left, select SSL under the Protocol group
3. Check the Enable SSL checkbox
4. Check the Support SSL intercepting checkbox
5. Upload root certificate
6. Upload root key
7. Apply changes and restart the device

4.4.2 Symantec Blue Coat Proxy SG

Prerequisite
- 3. Configuring MetaDefender ICAP Server
- Enable persistent connections (recommended) - Blue Coat is reusing connections to the ICAP server, so it is highly recommended to enable persistent connections on the ICAP side or the Blue Coat might detect some ICAP connection drop errors under high load.

ProxySG Management Console

The ProxySG configuration should be done from the ProxySG Management Console interface. Below is the minimum configuration required for MetaDefender ICAP Serverscan ICAP integration with ProxySG. Please refer to the ProxySG manual for advanced proxy configuration. Open a web browser and load the ProxySG Management Console. (Please refer to the ProxySG manual for details about how to open the ProxySG Management Console.) The ProxySG configuration should be done from the ProxySG Management Console interface. Below is the minimum configuration required for MetaDefender Core ICAP integration with ProxySG. Please refer to the ProxySG manual for advanced proxy configuration. Open a web browser and load the ProxySG Management Console. (e.g. https://<ip address>:8082 Please refer to the ProxySG manual for details about how to open the ProxySG Management Console.)
**Disable Automatic Cache Refresh**

1. Click on the 'Configuration' tab, and navigate to 'Proxy Settings'->'HTTP Proxy'
2. Select the 'Freshness' tab and select the 'Disable refreshing' option
3. Select the 'Acceleration Profile' tab and uncheck the following options
   a. Pipeline embedded objects in client request
   b. Pipeline redirects for client request
   c. Pipeline embedded objects in prefetch request
   d. Pipeline redirects for prefetch request
4. Click 'Apply' to save these settings

**Adding REQMOD Service (Upload Mode)**

1. Within the 'Configuration' tab, navigate to 'External Services'->'ICAP'
2. Click 'New'
3. Enter a service name for the Metascan service (in this example we use 'MetascanReqmod') and click 'OK'
4. In the services list, select 'MetascanReqmod' and click 'Edit'
5. Update the following values
   a. In ICAP Service
      i. Set Service URL to 'icap://<Metascan Server>/OMSScanReq-AV'
      ii. Select 'Use vendor's "virus found" page'
   b. In ICAP Service Ports
      i. Check 'This service supports plain ICAP connections'
      ii. Set the 'Plain ICAP port' value to your Metascan's ICAP port (1344 by default)
   c. In ICAP v1.0 Options
      i. Check 'Request modification'
      ii. Check 'Send Client address'
6. Click 'OK'
7. Click 'Apply' to save the configuration
Adding RESP MOD Service (Download Mode)

1. Within the 'Configuration' tab, navigate to 'External Services'->'ICAP'
2. Click 'New'
3. Enter a service name for the Metascan service (in this example we use 'MetascanRespmod') and click 'OK'
4. In the services list, select 'MetascanReqmod' and click 'Edit'
5. Update the following values
   a. In ICAP Service
      i. Set Service URL to 'icap://<Metascan Server>/OMSScanResp-AV'
      ii. Select 'Use vendor's "virus found" page'
   b. In ICAP Service Ports
      i. Check 'This service supports plain ICAP connections'
      ii. Set the 'Plain ICAP port' value to your Metascan's ICAP port (1344 by default)
   c. In ICAP v1.0 Options
      i. Check 'Response modification'
      ii. Check 'Send Client address'
6. Click 'OK'
7. Click 'Apply' to save the configuration

Create MetaDefender REQMOD Policy

1. Within the 'Configuration' tab, navigate to 'Policy'->'Visual Policy Manager'
2. Click the 'Launch' button
3. In the 'Blue Coat Visual Policy Manager' window, navigate to 'Policy'->'Add Web Content Layer'
4. Enter a layer name (in this example we use 'MetaDefender ICAP Server ReqMod') and click 'OK'
5. In the newly created 'MetaDefender ICAP Server ReqMod' tab, right click on 'Use Default Caching' and choose 'Set...'
6. In the 'Set Action Object' window, click 'New' and select 'Set ICAP Request Service...'
7. In the 'Add ICAP Request Service Object' window, set the following values
   a. Set 'name' to 'MetaDefender ICAP Server Request Service'
   b. In 'Available services', select 'MetascanReqMod' and click 'Add'

8. Click 'OK' to finish and 'Apply' to save

Create MetaDefender RESPMOD Policy

1. Within the 'Configuration' tab, navigate to 'Policy'->'Visual Policy Manager'
2. Click the 'Launch' button
3. In the 'Blue Coat Visual Policy Manager' window, navigate to 'Policy'->'Add Web Content Layer'
4. Enter a layer name (in this example we use 'MetaDefender ICAP Server RespMod') and click 'OK'
5. In the newly created 'MetaDefender ICAP Server RespMod' tab, right click on 'Use Default Caching' and choose 'Set...'
6. In the 'Set Action Object' window, click 'New' and select 'Set ICAP Response Service...' 
7. In the 'Add ICAP Response Service Object' window, set the following values
   a. Set 'name' to 'MetaDefender ICAP Server Response Service'
   b. In 'Available services', select 'MetascanRespMod' and click 'Add'
8. Click 'OK' to finish and 'Apply' to save

Configure Blue Coat SSL

refer to Configure Symantec Blue Coat ProxySG SSL

Bypassing MetaDefender ICAP Server temporarily

This guide describes how to enable and disable ICAP server on Symantec Blue Coat ProxySG.

This documentation is assuming that MetaDefender ICAP Server is already configured and working properly.

Enable MetaDefender ICAP Server

To enable MetaDefender ICAP Server on Blue Coat:

1. Open the Blue Coat Management Console.
2. Go to "Configure" tab > "Advanced configuration" button.
3. In the left side menu, go to "Policy" > "Visual Policy Manager"
4. Click "Launch"
5. In the VPM window, right click on the service you want to enable tab (typically "ICAP Respmod" or "ICAP Reqmod")
6. Click "Enable layer".
   - The tab color should turn black.
7. Click "Install policy"

---

**Disable MetaDefender ICAP Server**

To disable ICAP server on Blue Coat:

1. Open the Blue Coat Management Console.
2. Go to "Configure" tab > "Advanced configuration" button.
3. In the left side menu, go to "Policy" > "Visual Policy Manager"
4. Click "Launch"
5. In the VPM window, right click on the service you want to disable tab (typically "ICAP Respmod" or "ICAP Reqmod")
6. Click "Disable layer".
   - The tab color should turn red.
7. Click "Install policy"
Check cached objects properties
This article describes how to check if an object is cached by Symantec Blue Coat ProxySG and its current properties.

Prerequisites
1. ICAP is setup with Blue Coat.
2. Blue Coat caching is enabled (default).
   • See: Disable Symantec Blue Coat ProxySG caching

Step-by-step guide
Use the following syntax to check if an object is in cache:
https://<proxyip>:8082/CE/Info/<protocol name,
e.g. http>/<url without the protocol specification, e.g. without http://>
Example:
The check if the following object is in the cache:
http://www.bluecoat.com/sites/default/themes/bc_pandemonium/css/docviewer.css
Enter the following in the browser:
https://1.2.3.4:8082/CE/Info/http/www.bluecoat.com/sites/default/themes/bc_pandemonium/css/docviewer.css
If the object is in the cache, you will obtain a table providing details on the cached object such as:

Related articles
Error rendering macro 'contentbylabel' : parameters should not be empty
Configure Symantec Blue Coat ProxySG SSL

Enabling Blue Coat To Intercept SSL traffic
By default SSL (HTTPS) connections are not intercepted by Blue Coat and therefore data in them are not scanned by the MetaDefender ICAP Server. If you would like to scan files which were sent using secure connection, then you can optionally configure Blue Coat to decrypt SSL connections.

How To Configure
Please refer to Blue Coat documentation.

Limitations

- If the ICAP server is not connected directly to Blue Coat or it is not in a private network, then the connection between Blue Coat and ICAP won't be secure anymore and the decrypted data could be in danger. ([https://symwisedownload.symantec.com/resources/sites/SYMWISE/content/live/DOCUMENTATION/11000/DOC11474/en_US/SGOSAdmin71.pdf](https://symwisedownload.symantec.com/resources/sites/SYMWISE/content/live/DOCUMENTATION/11000/DOC11474/en_US/SGOSAdmin71.pdf) "Securing access to an ICAP Server")

- Valid SSL certificates are needed for Blue Coat and user experience could be altered by certification notifications.

How to overcome certificate issues

1. When creating a keyring and certificate explained in the Blue Coat documentation please give attention to that the **Common name** "must match the ProxySG name or IP address that the client expects"

2. After the keyring and the certificate is ready go to **Statics → Advanced → SSL → Download a ProxySG Certificate as a CA certificate** in ProxySG Management Console

3. Select the previously created certificate and download/install it to the browser in use

4. This certificate should be set under **Proxy Settings → SSL Proxy** and under the **SSLInterception** which was created during configuring SSL interception
Data Trickling

Overview
Blue Coat ProxySG appliances implement Data Trickling to improve the user experience during ICAP scanning. Internet Content Adaptation Protocol (ICAP) is the protocol used by Blue Coat ProxySG and ProxyAV appliances, as well as some third party partner appliances, to perform scanning of objects to detect viruses, worms, spyware, and Trojans. Data Trickling is a mechanism implemented by Blue Coat ProxySG appliances performing ICAP scanning that slowly delivers, or trickles, data to the client as it is being scanned. By trickling data, users do not experience the timeouts sometimes associated with waiting for large objects to be scanned, or when scanning is delayed by high loads on content servers or upstream bandwidth limitations.

How does Data Trickling work?
Data Trickling is designed to prevent the timeouts that can sometimes be associated with patience pages. To prevent such timeouts, Data Trickling trickles – or transmits at a very slow rate – bytes to the client at the beginning of the scan or near the very end. Because the ProxySG appliance begins serving content without waiting for the ICAP scan result, timeouts do not occur. However, to maintain security, the full object is not delivered until the results of the content scan are complete (and the object is determined to not be infected). Two types of Data Trickling are available on Blue Coat ProxySG appliances – trickle from start and trickle at end.

Trickle from start
In trickle from start mode, the ProxySG appliance buffers a small amount of the beginning of the response body. As the ICAP server continues to scan the response, the ProxySG appliance allows one byte per second to the client. After the ICAP server completes its scan, if the object is deemed to be clean (no response modification is required), the ProxySG appliance sends the rest of the object bytes to the client at the best speed allowed by the connection. If the object is deemed to be malicious, the ProxySG appliance terminates the connection and the remainder of the response object. Trickling from the start is the more secure Data Trickling option because the client receives only a small amount of data pending the outcome of the virus scan.

Trickle at end
In trickle at end mode, the ProxySG appliance sends the response to the client at the best speed allowed by the connection, except for the last 16KB of data. As the ICAP server performs the content scan, the ProxySG appliance allows one byte per second to the client. After the ICAP server completes its scan, if the object is deemed to be clean (no response modification is required), the ProxySG appliance sends the rest of the object bytes to the client.
at the best speed allowed by the connection. This method is more user-friendly than trickle at start. This is because users tend to be more patient when they notice that 99% of the object is downloaded versus 1%, and are less likely to perform a connection restart. However, network administrators might perceive this method as the less secure method, as a majority of the object is delivered before the results of the ICAP scan.

**Step-by-step guide**

To enable data trickling:

1. Open the BlueCoat Management Console.
2. Go to "Configuration" tab > "Advanced configuration" button.
3. Enter credentials if prompted.
4. In the Advanced configuration menu, go to "Configuration" tab > "External Services" > "ICAP".
5. Click the "ICAP Feedback" tab.
6. In the "ICAP Feedback for Interactive Traffic" section:
   a. Check "Provide feedback after X seconds" checkbox
   b. Set the number of seconds to the time you want to wait for ICAP to respond before starting trickling
      - 8 seconds is a usually a good timing, long enough for average file sizes to be fully scanned by ICAP, short enough for browsers to not timeout before trickling starts.
   c. Check the "Trickle object data from start" or "Trickle object data at end" depending on the trickling type you want (see "How does Data Trickling work" section).
      - "From start" is the most secure.
      - "At end" is the most user friendly.
7. In the "ICAP Feedback for Non-Interactive Traffic" section:
   a. Check "Provide feedback after X seconds" checkbox
   b. Set the number of seconds to the time you want to wait for ICAP to respond before starting trickling
      - 5 seconds is a usually a good timing for non-interactive traffic
   c. Check the "Trickle object data from start" or "Trickle object data at end" depending on the trickling type you want (see "How does Data Trickling work" section).
      - "From start" is the most secure.
      - "At end" is the most user friendly.
Disable Symantec Blue Coat ProxySG caching

Step-by-step guide

To prevent the transmission of stale content or other issues caused by object caching, you can use either `cache(no)` or `bypass_cache(yes)` in content policy language (CPL). For a comparison of `cache(no)` and `bypass_cache(yes)`, see KB1772.

This sample provides instructions for disabling object caching for specific URLs by adding a policy rule in the Web Access Layer.

1. In the Management Console, select **Configuration > Policy > Visual Policy Manger**, and then click **Launch**.
3. In the Add New Layer dialog box, name the layer to reflect the purpose of the layer, such as "Web Access Layer (bypass cache)," then click **OK**.
4. Right click the **Destination** field and select **Set** from the drop-down list.
5. In the Set Destination Object dialog box, click **New > Request URL**, and enter the URL you want to exclude from the cache. Click **Add**, then **OK**.
6. Right click on **Action** and select **Set > Bypass Cache**. Then, click **OK**.
7. Click **Install policy** to apply the new policy.

To use CPL code for the same policy rules, add the following CPL code in the local policy file or in a VPM CPL Policy Layer:

```
<Proxy>
url.domain=<url> bypass_cache(yes)
```

In the example above, `<url>` is the URL you want to exclude from caching.

**Source:**
Related articles

Error rendering macro 'contentbylabel' : parameters should not be empty

Process Only Specific File Types

Step-by-step guide

1. Open the Symantec Blue Coat ProxySG management console using a web browser (default URL: http://<bluecoat proxy IP>:8082/)

2. When prompted, enter your user name and password and click "Ok".

3. In the Blue Coat management console, click on the "Configure" tab) 

![Blue Coat ProxySG 810-20](image)

        Report Configure System Settings

4. On the "Configuration" page, click on the "Advanced configuration" button
5. In the management console menu, go to Policy -> Visual Policy Manager.
6. Click "Launch" to start the Visual Policy Manager.

7. In the Visual Policy Manager, click on the tab where you set the ICAP rule you plan to add file type filtering on.

- In these instructions, the ICAP layer is named "Metascan ICAP RespMod". The tab name on your installation might be different.
- The window you see should look similar to the screenshot below.
8. Right click the "Destination" cell of the ICAP rule (line where the "Action" is set to your ICAP service) and click "Set".

9. In the "Set Destination Object", select "New" and click "Apparent Data Type".
10. In the "Add Apparent Data Type Object" window, change the "Name" value to whatever you want (optional) and select the file types you want forwarded to the ICAP server.
11. Click "Ok" twice to get back to the Visual Policy Manager.

12. In the "Visual Policy Manager" window, click "Install policy" to apply the new configuration.
4.4.3 F5 Integrations

MetaDefender ICAP Server can be integrated to different F5 solutions in different ways. Please follow the guide which is the best match for your use-case:

- MetaDefender ICAP Server and BIG-IP LTM using iApp template: ICAP Server iApp template
- MetaDefender ICAP Server and BIG-IP LTM without using iApp template: F5 BIG IP LTM
- MetaDefender ICAP Server and BIG-IP ASM (WAF): F5 BIG IP ASM (WAF)
- MetaDefender ICAP Server and SSL Orchestrator: F5 SSL Orchestrator
- Configuring SSL interception: F5 BIG IP SSL configuration

F5 BIG IP ASM (WAF)

- System Requirements
- Configuring MetaDefender ICAP Server
  - Note on MetaDefender ICAP Server Licensing
- Configuring the F5 BIG IP Appliance
• Configuring the ICAP server
• Configuring the antivirus blocking settings
• Configuring the internal system variables
• Configuring antivirus scanning for HTTP file uploads and SOAP attachments

• Testing the configuration
• Viewing reports for anti-virus detection

Using F5’s Web Application Firewall solution together with MetaDefender ICAP Server will help you prevent malware reaching your web application by analyzing files being uploaded with all of the engines in MetaDefender Core. You can also use MetaDefender’s CDR technology by setting up MetaDefender ICAP Server to your existing virtual server using MetaDefender’s iApp template. This is a step by step guideline on how to enable threat prevention functionality for F5 BIG IP ASM by leveraging MetaDefender ICAP Server.

System Requirements

The following systems are required to set up MetaDefender ICAP Server with an F5 BIG IP

• F5 BIG IP with ASM
• MetaDefender ICAP Server
• MetaDefender Core

Configuring MetaDefender ICAP Server

For installation and configuration quick guidelines see: 1. Quick Start with MetaDefender ICAP Server.
For detailed instructions see 2. Installing or Upgrading MetaDefender ICAP Server and 3. Configuring MetaDefender ICAP Server.

**Note on MetaDefender ICAP Server Licensing**

MetaDefender ICAP Server must have a valid license to function correctly. For license configuration details see 2.4. MetaDefender ICAP Server Licensing.

**Configuring the F5 BIG IP Appliance**

The following configuration steps should be done from the F5 BIG IP Management Console interface. The steps below describe the minimum configuration required for MetaDefender ICAP Server integration with F5 BIG IP. Please refer to Configuring BIG-IP ASM antivirus protection for a more advanced configuration.

Open a web browser and load the BIG IP Management Console. (Please refer to the BIG IP manual for details about how to open the BIG IP Management Console.)

This guide was written with a presumption that there is an already existing Virtual Server with an active Security Policy in the BIG IP ASM which we would like to protect against viruses.

**Configuring the ICAP server**

1. Navigate to **Security > Options > Application Security > Integrated Services > Anti-Virus Protection**.

2. Enter the ICAP server hostname or IP address in the **Server Host Name/IP Address** field.

3. Enter the ICAP server port in the **Server Port Number** field or leave the default value of 1344.

4. Select the **Guarantee Enforcement** option if you want the system to perform virus checking even if performing checking may slow your web application.

5. Click **Save**.

6. To activate the security policy changes immediately, click **Apply Policy**.
Configuring the antivirus blocking settings

The information here is applicable for BIG-IP version 13.x. For other versions please check Configuring BIG-IP ASM antivirus protection.


2. Expand Antivirus Protection and select either or both of the Alarm and Block check boxes for the Virus Detected violation.

3. Click Save.

4. To activate the security policy changes immediately, click Apply Policy.
Configuring the internal system variables

You can configure the ICAP URI and the virus header name settings on the System Variables page in the Configuration utility. MetaDefender ICAP Server works with the default values out of the box so you don't need to configure them. If you would like to configure them anyway just follow these steps:


2. For the icap_uri setting, enter the URI for the ICAP service, which checks requests for viruses by connecting to ICAP server. You can keep the default value ("/reqmod") for MetaDefender ICAP Server.

3. For the virus_header_name setting, enter the header name used by an anti-virus program on an ICAP server. For MetaDefender ICAP Server the default headers (X-Virus-Name, X-Infection-Found) can be used as it supports X-Infection-Found. Here you can find the headers supported by MetaDefender ICAP Server.

4. Click Save.
Configuring antivirus scanning for HTTP file uploads and SOAP attachments


2. Select the relevant security policy from the list

3. Click the Inspect file uploads within HTTP requests check box.

4. To perform antivirus scanning on SOAP attachments, move the relevant XML profiles from the Antivirus Protection Disabled list to the Antivirus Protection Enabled list.

5. Click Save.

6. To activate the security policy changes immediately, click Apply Policy.

[Image of the user interface showing the configuration page]

Testing the configuration

After you have finished the steps above your web application should be protected against malicious file uploads. To check this navigate to a page in your application where you can upload files and try to upload an eicar test file.

For example a really simple file upload application could look like this:
When trying to submit the file upload the request will be blocked and the default block page will be displayed which should look like this:

```
The requested URL was rejected. Please consult with your administrator.
Your support ID is: 13106536790477504544
[Go Back]
```

For configuring the block page please refer to Configuring what happens if a request is blocked.

You can also check the ICAP history to see that the processing took place:
Viewing reports for anti-virus detection

To view reports of virus transactions detected by the ASM system, perform the following steps:

1. Navigate to **Security > Reporting > Applications > Charts**.
2. Select **Top Viruses Detected** from the drop down menu.
3. The system displays the detected viruses over time.

---

**F5 BIG IP LTM**

- Sample Deployment Diagram
- System Requirements
- Deployment ‘How To’ Video
- Configuring MetaDefender ICAP Server
  - Note on MetaDefender ICAP Server Licensing
- Configuring the F5 BIG IP Appliance
  - Creating a custom client-side ICAP profile
  - Creating a pool of ICAP servers
  - Creating an internal virtual server for forwarding requests to an ICAP server
Creating a custom Request Adapt profile
Creating a custom HTTP profile
Creating a pool to process HTTP traffic
Creating an HTTP virtual server for enabling request adaptation
Configuring the REQMOD (Request Modification) service
Configuring REQMOD and RESPMOD Services
Configuring Service Down Actions
- Bypass ICAP server on service down
- Transfer content to different pool member

- Throughput limitation by license

MetaDefender Core—via MetaDefender ICAP Server—can be used to scan and sanitize all files being uploaded to the F5 BIG IP server with all of the engines in MetaDefender Core to make sure that no malware is able to get to the web servers behind the Big IP server. This guide describes the basic steps to getting MetaDefender ICAP Server working with your F5 BIG IP server.

The simplest way to integrate MetaDefender ICAP Server with F5 BIG IP is to use the iApps framework of F5.

The iApps technology was released in F5’s v11, so the description in this page is more generic.

For details about the MetaDefender ICAP Server iApp template and about how to use it, see ICAP Server iApp template.
System Requirements
The following systems are required to set up MetaDefender ICAP Server with an F5 BIG IP

- F5 BIG IP with LTM
- MetaDefender ICAP Server
- MetaDefender Core

Deployment ‘How To’ Video
To help you get up and running quickly, watch this video that demonstrates how to integrate MetaDefender ICAP Server with F5® BIG-IP® Load Traffic Manager™ (LTM®): https://www.opswat.com/videos/how-to-integrate-metadefender-icap-with-f5

Configuring MetaDefender ICAP Server
For installation and configuration quick guidelines see: 1. Quick Start with MetaDefender ICAP Server.
For detailed instructions see 2. Installing or Upgrading MetaDefender ICAP Server and 3. Configuring MetaDefender ICAP Server.

Note on MetaDefender ICAP Server Licensing

MetaDefender ICAP Server must have a valid license to function correctly. For license configuration details see 2.4. MetaDefender ICAP Server Licensing.

Configuring the F5 BIG IP Appliance

The following configuration steps should be done from the F5 BIG IP Management Console interface. The steps below describe the minimum configuration required for MetaDefender ICAP Server integration with F5 BIG IP. Please refer to the F5 BIG IP manual for advanced configuration.

1. Open a web browser and load the BIG IP Management Console. (Please refer to the BIG IP manual for details about how to open the BIG IP Management Console.)

Creating a custom client-side ICAP profile

You create this ICAP profile when you want to use an ICAP server to wrap an HTTP request in an ICAP message before the BIG-IP system sends the request to a pool of web servers. The profile specifies the HTTP request-header values that the ICAP server uses for the ICAP message. Important: You can use macro expansion for all ICAP header values. For example, if an ICAP header value contains ${SERVER_IP}, the BIG-IP system replaces the macro with the IP address of the ICAP server selected from the pool assigned to the internal virtual server. If an ICAP header value contains ${SERVER_PORT}, the BIG-IP system replaces the macro with the port of the ICAP server selected from the pool assigned to the internal virtual server. For example, you can set the URI value in an ICAP profile to icap://${SERVER_IP}:${SERVER_PORT}/OMSScanReq-AV.

1. On the Main tab, click Local Traffic > Profiles > Services > ICAP.
2. Click Create.

3. In the Name field, type a unique name for the profile.

4. For the Parent Profile setting, retain the default value, icap.

5. On the right side of the screen, select the Custom check box.

6. In the URI field, type a URI in this format: icap://hostname:port/path. For example, using macro expansion, you can set the URI value to:icap://${SERVER_IP}:${SERVER_PORT}/OMSScanReq-AV.

7. In the Preview Length field, type a length or retain the default value 0. This value defines the amount of the HTTP request or response that the BIG-IP system offers to the ICAP server when sending the request or response to the server for adaptation. This value should not exceed the length of the preview that the ICAP server has indicated it will accept.

8. Leave empty for "Header From", "Host", "Referer", "User Agent" fields.

9. Click Finished.
After you create the ICAP profile, you can assign it to an internal virtual server so that the HTTP request that the BIG-IP system sends to an ICAP server is wrapped in an ICAP message, according to the settings you specified in the ICAP profile.

Creating a pool of ICAP servers

You perform this task to create a pool of ICAP servers that perform content adaptation on HTTP requests.

2. Click Create. The New Pool screen opens.
3. In the Name field, type a unique name for the pool.
4. For the Health Monitors setting, from the Available list, select a monitor other than http(s), and click << to move the monitor to the Active list.
5. From the Load Balancing Method list, select how the system distributes traffic to members of this pool. The default is Round Robin.
6. For the Priority Group Activation setting, specify how to handle priority groups:
   - Select Disabled to disable priority groups. This is the default option.
   - Select Less than, and in the Available Members field type the minimum number of members that must remain available in each priority group in order for traffic to remain confined to that group.
7. Using the New Members setting, add each resource that you want to include in the pool:
   a. Either type an IP address in the Address field, or select a node address from the Node List.
   b. Type a port number in the Service Port field, or select a service name from the list.
   c. To specify a priority group, type a priority number in the Priority field.
   d. Click Add.
8. Click Finished.

The pool of ICAP load balancing servers appears in the Pools list.

Creating an internal virtual server for forwarding requests to an ICAP server

A virtual server of type internal provides a destination that a standard type of virtual server can use when forwarding HTTP requests slated for ICAP-based content adaptation.

2. Click the Create button. The New Virtual Server screen opens.

3. In the Name field, type a unique name for the virtual server.

4. In the Description field, type a description of the virtual server. For example: This virtual server ensures HTTP request modification through the use of the service_name ICAP service..

5. From the Type list, select Internal.

6. For the State setting, verify that the value is set to Enabled.

7. From the Configuration list, select Advanced.

8. From the ICAP Profile list, select the ICAP profile that you previously created for handling HTTP requests.

9. From the Default Pool list, select the pool of ICAP servers that you previously created.

10. Click Finished.

After you perform this task, a standard type of virtual server can forward HTTP requests to an internal type of virtual server. The internal virtual server then sends the request to a pool of ICAP servers, before sending the request back to the standard virtual server for forwarding to the pool of web servers.

Creating a custom Request Adapt profile

You create a Request Adapt type of profile when you want a standard HTTP virtual server to forward HTTP requests to an internal virtual server that references a pool of ICAP servers. A Request Adapt type of profile instructs the HTTP virtual server to send an HTTP request to a named internal virtual server for possible request modification.

1. On the Main tab, click Local Traffic > Profiles > Services > Request Adapt.

2. Click Create.

3. In the Name field, type a unique name for the profile.

4. For the Parent Profile setting, retain the default value, requestadapt.

5. On the right side of the screen, click the Custom check box.

6. For the Enabled setting, retain the default value, Enabled. When you set this value to Enabled, the BIG-IP system forwards HTTP requests to the specified internal virtual server for adaptation.

7. From the Internal Virtual Name list, select the name of the internal virtual server that you previously created for forwarding HTTP requests to the pool of iCAP servers.
8. In the Preview Size field, type a numeric value. This specifies the maximum size of the preview buffer. This buffer holds a copy of the HTTP request header and the data sent to the internal virtual server, in case the adaptation server reports that no adaptation is needed. Setting the preview size to 0 disables buffering of the request and should only be done if the adaptation server always returns a modified HTTP request or the original HTTP request.

9. In the Timeout field, type a numeric value, in seconds. If the internal virtual server does not return a result within the specified time, a timeout error occurs. To disable the timeout, use the value 0.

10. From the Service Down Action list, select an action for the BIG-IP system to take if the internal virtual server returns an error:
   - Select Ignore to instruct the BIG-IP system to ignore the error and send the unmodified HTTP request to an HTTP server in the HTTP server pool.
   - Select Drop to instruct the BIG-IP system to drop the connection.
   - Select Reset to instruct the BIG-IP system to reset the connection.

11. Click Finished.

After you perform this task, the BIG-IP system contains a Request Adapt profile that a standard HTTP virtual server can use to forward an HTTP request to an internal virtual server for ICAP traffic.

Creating a custom HTTP profile

An HTTP profile defines the way that you want the BIG-IP® system to manage HTTP traffic. Note: Other HTTP profile types (HTTP Compression and Web Acceleration) enable you to configure compression and cache settings, as required. Use of these profile types is optional.

1. On the Main tab, click Local Traffic > Profiles > Services > HTTP. The HTTP profile list screen opens.
2. Click Create. The New HTTP Profile screen opens.
3. In the Name field, type a unique name for the profile.
4. From the Parent Profile list, select http.
5. Select the Custom check box.
6. Modify the settings, as required.
7. Click Finished.

The custom HTTP profile now appears in the HTTP profile list screen.
Creating a pool to process HTTP traffic

You can create a pool of web servers to process HTTP requests.

2. Click Create. The New Pool screen opens.
3. In the Name field, type a unique name for the pool.
4. For the Health Monitors setting, from the Available list, select a monitor other than http(s), and click << to move the monitor to the Active list.
5. From the Load Balancing Method list, select how the system distributes traffic to members of this pool. The default is Round Robin.
6. For the Priority Group Activation setting, specify how to handle priority groups:
   - Select Disabled to disable priority groups. This is the default option.
   - Select Less than, and in the Available Members field type the minimum number of members that must remain available in each priority group in order for traffic to remain confined to that group.
7. Using the New Members setting, add each resource that you want to include in the pool:
   a. Type an IP address in the Address field, or select a node address from the Node List.
   b. Type 80 in the Service Port field, or select HTTP from the list.
   c. (Optional) Type a priority number in the Priority field.
   d. Click Add.
8. Click Finished.

The new pool appears in the Pools list.

Creating an HTTP virtual server for enabling request adaptation

You perform this task to create a standard virtual server that can forward an HTTP request to an internal virtual server. The internal virtual server then sends the request to a pool of ICAP servers before the BIG-IP® system sends the request to the web server.

2. Click the Create button. The New Virtual Server screen opens.
3. In the Name field, type a unique name for the virtual server.
4. For the Destination setting, in the Address field, type the IP address that you want to use as a destination for client traffic destined for a pool of HTTP web servers. The IP address you type must be available and not in the loopback network.

5. In the Service Port field, type 80, or select HTTP from the list.

6. From the Configuration list, select Advanced.

7. From the HTTP Profile list, select the name of the HTTP profile that you created previously.

8. From the Request Adapt Profile list, select the name of the Request Adapt profile that you previously created.

9. From the Source Address Translation list, select Auto Map.

10. From the Default Pool list, select the name of the HTTP server pool that you previously created.

11. Click Finished.

Configuring the REQMOD (Request Modification) service

In order to configure F5 BIG-IP LTM to only forward HTTP requests to the MetaDefender Core ICAP server, follow the steps described below. In the case you want to configure F5 BIG-IP LTM to forward both HTTP requests and responses, refer to the "Configuring REQMOD and RESPMOD Services" section.


2. Update the REQMOD ICAP service profile.
   a. Go to “Local Traffic” > “Profiles” > “Services” > “ICAP”.
   b. In the list that appears select your ICAP Request mod service.
   c. Set “Preview Length” to 0 and make sure the checkbox next to it is checked.
   d. Click “Update” to apply the changes.

3. Update the Request Adapt profile.
   a. Go to “Local Traffic” > “Profiles” > “Services” > “Request Adapt”.
   b. In the list that appears select your request adapt service.
   c. Set “Preview Size” to 0 and make sure the checkbox next to it is checked.
   d. Click “Update” to apply the changes.
Configuring REQMOD and RESP MOD Services

In order to configure F5 BIG-IP LTM to forward both HTTP requests and responses to the MetaDefender Core ICAP server, follow the steps described below. In the case you want to configure F5 BIG-IP LTM to only forward HTTP responses, refer to the "Configuring REQMOD Service" section.

1. Open a Web browser and follow the instructions from the page:

2. Update your REQMOD ICAP service profile.
   a. Go to “Local Traffic” > “Profiles” > “Services” > “ICAP”.
   b. In the list that appears select your ICAP Request mod service.
   c. Set “Preview Length” to 0 and make sure the checkbox next to it is checked.
   d. Click “Update” to apply the changes.

3. Update your RESP MOD ICAP service profile.
   a. Go back to “Local Traffic” > “Profiles” > “Services” > “ICAP”.
   b. In the list that appears select your ICAP Response mod service.
   c. Set “Preview Length” to 0 and make sure the checkbox next to it is checked.
   d. Click “Update” to apply the changes.

4. Update your Request Adapt profile.
   a. Go to “Local Traffic” > “Profiles” > “Services” > “Request Adapt”.
   b. In the list that appears select your request adapt service.
   c. Set “Preview Size” to 0 and make sure the checkbox next to it is checked.
   d. Click “Update” to apply the changes.

5. Update Response Adapt service profile (only if RESP MOD is used)
   a. Go to “Local Traffic” > “Profiles” > “Services” > “Response Adapt”.
   b. In the list that appears select your response adapt service.
   c. Set “Preview Size” to 0 and make sure the checkbox next to it is checked.
   d. Click “Update” to apply the changes.

Configuring Service Down Actions

If you followed the steps described in "Configuring REQMOD Service" or "Configuring REQMOD and RESP MOD Services". Big-IP will be configured to drop all connections when the ICAP service is down.
F5 can be configured to forward HTTP data to the web server/web client in the case the ICAP server is unreachable. If you are using an ICAP server pool that contains more than one MetaDefender ICAP Server, F5 can also be configured to forward the HTTP content to a different pool member.

**Bypass ICAP server on service down**

Note that bypassing ICAP on service down may lower your organisation's security as content will be forwarded without being scanned.

1. Open the “Request adapt” profile (“Profiles” > “Services” > “Request Adapt”)
2. Set “Service Down Action” to “Ignore”.
3. Click the "Update" button to apply the changes.

**Transfer content to different pool member**

If you are using an ICAP server pool that contains more than one MetaDefender ICAP Server, you can also configure Big-IP to send the HTTP content to a different ICAP pool member.

1. Open your ICAP services pool properties (“Pools” > “Pool List”).
2. Set the "Configuration" list to "Advanced".
3. Set the “Action on Service Down” property to “Reselect”.
4. Click the "Update" button to apply the changes

**Throughput limitation by license**

If you experience slow download/upload through F5 then there is a chance that your throughput is limited by F5 license.

How to check the maximum throughput allowed by license:

1. SSH into F5: On Windows open PuTTY then type the IP of the F5 device, and click Open
2. Use the default login: admin/admin

3. Type tmsh and press enter

4. Type "show /sys license detail | grep perf" to see performance limitations by license

5. To exit from tmsh type "quit" and press enter, to quit from PuTTY type "exit" then press enter

**F5 BIG IP SSL configuration**

To be able to scan data transmitted using SSL connection you have to take similar steps as listed in **F5 BIG IP LTM**.

The only difference is that you should setup a HTTPS pool and virtual server instead of plain HTTP.

Please check the following links to be able to setup HTTPS connection handling:

- Managing client-side HTTPS traffic using a self-signed certificate
- Managing client and server HTTPS traffic using a self-signed certificate
- Managing client-side HTTPS traffic using a CA-signed certificate

**F5 SSL Orchestrator**

- System Requirements
Using F5's SSL Orchestrator together with MetaDefender ICAP Server will help you scanning your non-SSL and decrypted SSL traffic flow for threats using all of the engines in MetaDefender Core. This guide describes the basic steps to getting MetaDefender ICAP Server working with your F5 SSL Orchestrator.

**System Requirements**

The following systems are required to set up MetaDefender ICAP Server with an F5 SSL Orchestrator

- F5 SSL Orchestrator
- MetaDefender ICAP Server
- MetaDefender Core

**Configuring MetaDefender ICAP Server**

For installation and configuration quick guidelines see: 1. *Quick Start with MetaDefender ICAP Server*.

For detailed instructions see 2. *Installing or Upgrading MetaDefender ICAP Server* and 3. *Configuring MetaDefender ICAP Server*.

**Note on MetaDefender ICAP Server Licensing**

MetaDefender ICAP Server must have a valid license to function correctly. For license configuration details see 2.4. *MetaDefender ICAP Server Licensing.*
Configuring F5 SSL Orchestrator

The following configuration steps should be done from the F5 SSL Orchestrator Management Console interface. The steps below describe the minimum configuration required for MetaDefender ICAP Server integration with F5 SSL Orchestrator and was created based on SSL Orchestrator v3.0.

Open a web browser and load the SSL Orchestrator Management Console. (Please refer to the SSL Orchestrator manual for details about how to open the BIG IP Management Console.)

This guide was written with a presumption that you have already completed the General Properties configuration in the SSL Orchestrator.

Configuring the ICAP service

1. Navigate to **SSL Orchestrator > Configuration**
2. Select **ICAP / SWG Services** under the **Services** tab

![Diagram of SSL Orchestrator configuration]

3. Click **Add** to add a new ICAP service

![ICAP / SWG Services configuration table]

4. In the **Name** field, type a name for your configuration
5. Select **ICAP** as the **Service type**
6. Add your MetaDefender ICAP Server's IP and port to the **ICAP Devices**

7. Select the **Headers** mode. "Default" can be used.

8. Select **TCP Connections**. **OneConnect** can be used if you enabled persistent connections in your MetaDefender ICAP Server (enabled by default) otherwise use **Separate**.

9. Select "Load Balanced" as the **Type**. You can find more information about it [here](#).

10. You can use "reqmod" and "respmod" as the values for **Request** and **Response** fields

11. Select your **ICAP Policy** if you have any

12. Set 0 as the **Preview Max. Length**

13. Select your preferred **Server Failure Handling** (Next Service Chain or Reset Connection)

14. Select if you would like to send only HTTP/1.1 or both HTTP/1.0 and HTTP/1.1 requests to the ICAP service in **Send HTTP/1.0 Requests to ICAP**

15. Set your **Addition iRule** if you have any

16. Click **Finished**

17. Click **Save**

**Testing the configuration**

To check that you configuration is working as expected try to download an eicar testfile over HTTPS [here](#). If everything was setup properly you should see a blocking page similar to this:
4.7.3

ICAP Server iApp template

- iApps
  - iApps in a nutshell
  - Benefits for ICAP Server
- ICAP Server iApp template
  - Getting the template
    - Downloading the template
  - Importing the template into F5 BIG IP
  - Creating an ICAP Server application service
    - Basic configuration
  - Assigning request and response adapt profiles
  - Testing the configuration

⚠️ The iApps technology was released in F5’s v11.

For cases when iApps can not be used please see the generic integration guide: F5 BIG IP LTM.

iApps

iApps in a nutshell

iApps is a user-customizable framework for deploying applications that enables templatizing sets of functionality on an F5 gear. Automate the process of adding virtual servers or build a custom iApp to manage your iRules inventory. [F5-1]

iApps are powerful tools. Used by roughly one third of all F5 customers, they perform input validation and apply complex business logic for configuring a wide variety of applications. They hide complexity, sometimes driving hundreds of configuration parameters with just a handful of input values. They also provide deployment guidance, reducing the need for documentation and training. It is helpful to understand the difference between an iApp and a wizard. A wizard is usually a script with a GUI. It is a tool that accepts a set of user inputs and performs a one-time procedure. When run twice, a wizard performs exactly the same steps during its second run as it did during its first. Although an iApp is also a script with a GUI, it behaves differently on re-entry. Unlike a wizard, an iApp maintains a relationship to the configuration that it generates. [F5-2]

iApps have 5 critical properties: [F5-2]
• iApps always act atomically. The result of deployment is always either the entire intended configuration or none at all.

• iApp-driven configuration objects are marked so the iApp can track them throughout its lifecycle. A visualization of these configuration objects is presented in the popular TMUI “Component View.”

• iApps protect the configuration from accidental changes. iApp-driven elements may not be changed via the UI or CLI except through the iApp.

• iApps support re-entry. Since the iApp framework tracks the configuration objects that it manages, it can be intelligent about which elements are touched during reconfiguration.

• iApps automatically perform cleanup on deletion. The housekeeping is automatic, and the iApp author does not need to (and is not allowed to) write any delete-time code.

There is an analogy between iApps and iRules that help understanding iApps better. F5 products support a scripting language based on TCL. This language allows an administrator to tell their BIG-IP to intercept, inspect, transform, direct and track inbound or outbound application traffic. An iRule is the bit of code that contains the set of instructions the system uses to process data flowing through it, either in the header or payload of a packet. This technology allows our customers to solve real-time application issues, security vulnerabilities, etc that are unique to their environment or are time sensitive. [F5-3]

An iApp is like iRules, but for the management plane. Again, there is a scripting language that administrators can build instructions the system will use. But instead of describing how to process traffic, in the case of iApp, it is used to describe the user interface and how the system will act on information gathered from the user. The bit of code that contains these instructions is referred to as an iApp or iApp template. A system administrator can use F5-provided iApp templates installed on their BIG-IP to configure a service for a new application. They will be presented with the text and input fields defined by the iApp author. Once complete, their answers are submitted, and the template implements the configuration. First an application service object (ASO) is created that ties together all the configuration objects which are created, like virtual servers and profiles. Each object created by the iApp is then marked with the ASO to identify their membership in the application for future management and reporting. [F5-3]

Benefits for ICAP Server

From MetaDefender ICAP Server's perspective iApps is an exceptionally powerful tool to perform integration to F5 BIG IP in a quick and easy way, with no deeper knowledge about both BIG IP and ICAP Server.
ICAP Server iApp template

Getting the template

Downloading the template

The `opswat_metadefender_icap` iApp template can be downloaded from https://github.com/OPSWAT/F5-iApp.

Importing the template into F5 BIG IP

<table>
<thead>
<tr>
<th>Description</th>
<th>Screenshot</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 Login to F5 using an account with administrative privileges.</td>
<td><img src="image" alt="Login Screenshot" /></td>
</tr>
<tr>
<td>2</td>
<td></td>
</tr>
<tr>
<td>Description</td>
<td>Screenshot</td>
</tr>
<tr>
<td>----------------------------------------------------------------------------</td>
<td>------------</td>
</tr>
<tr>
<td>From the left side menu select <strong>Main &gt; iApps &gt; Templates</strong> and click the <strong>Import...</strong> button on the top of the <strong>Template List.</strong></td>
<td><img src="image_url" alt="Screenshot" /></td>
</tr>
</tbody>
</table>

3. Click the **Choose file** button and select the file containing the MetaDefender ICAP Server iApp template.
4 Click the *Upload* button and wait for the file being uploaded.
<table>
<thead>
<tr>
<th>Description</th>
<th>Screenshot</th>
</tr>
</thead>
<tbody>
<tr>
<td>5 Once the file gets uploaded the <code>opswat_metadefender_icap</code> template appears in the list (in a position defined by the current ordering of the list).</td>
<td><img src="image.png" alt="Screenshot" /></td>
</tr>
</tbody>
</table>
Creating an ICAP Server application service

Once the `opswat_metadefender_icap` iApp template is imported, it can be used to create new application services.

<table>
<thead>
<tr>
<th>Description</th>
<th>Screenshot</th>
</tr>
</thead>
<tbody>
<tr>
<td>1  From the left side menu select <em>Main &gt; iApps &gt; Application Services</em> and click the <em>Create...</em> button (or click the <code>i</code> icon next to <em>Application Services</em>).</td>
<td><img src="https://172.16.201.229/trmui/Control/jsp/map/trmui/application_template/properties.jsp?name=" alt="" /></td>
</tr>
</tbody>
</table>
2. From the *Template* list select *opswat_metadefender_icap*. 
<table>
<thead>
<tr>
<th>Description</th>
<th>Screenshot</th>
</tr>
</thead>
<tbody>
<tr>
<td>3 The fields of the <em>New Application Service</em>... window are populated</td>
<td><img src="image.png" alt="Screenshot of F5 iSeries" /></td>
</tr>
<tr>
<td>according to the selected template.</td>
<td></td>
</tr>
</tbody>
</table>
In this section we will create a basic configuration that works, but may not be adequate in a specific production environment.

⚠️ Do NOT select HTTP monitor for the ICAP pool

When selecting an HTTP monitor for the ICAP pool there will be a lot of "GET /" HTTP requests towards ICAP which will be displayed as a Bad Request in the ICAP history.
<table>
<thead>
<tr>
<th>Description</th>
<th>Screenshot</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>1</strong> Provide the <em>Name</em> for this application service.</td>
<td>![Screenshot of MetaDefender ICAP Server configuration]</td>
</tr>
<tr>
<td><strong>2</strong> Provide the <em>IP address</em> (and <em>Port</em>) of the MetaDefender ICAP Server. If an ICAP server node has previously been created then it can also be selected here.</td>
<td>![Screenshot of MetaDefender ICAP Server configuration]</td>
</tr>
</tbody>
</table>
If there are multiple ICAP servers, then any lot (at least one) can be added here.

<table>
<thead>
<tr>
<th>Description</th>
<th>Screenshot</th>
</tr>
</thead>
<tbody>
<tr>
<td>If there are multiple ICAP servers, then any lot (at least one) can be added here.</td>
<td><img src="image" alt="Screenshot" /></td>
</tr>
</tbody>
</table>

3 Click the *Finished* button.
<table>
<thead>
<tr>
<th>Description</th>
<th>Screenshot</th>
</tr>
</thead>
<tbody>
<tr>
<td>4. The configuration is created and displayed in the next screen.</td>
<td></td>
</tr>
<tr>
<td>Please note the <em>Metadefender_ICAP_Request</em> and <em>Metadefender_ICAP_Response</em> profiles created. These profiles</td>
<td></td>
</tr>
</tbody>
</table>
are needed to be assigned to a virtual server (see in the next section) so that content adaptation can work.

Assigning request and response adapt profiles

⚠️ As a final step the request and response adapt profiles –created in the previous section– are needed to be assigned to a virtual server so that content adaptation can work.
Creating virtual servers –besides creating ICAP virtual servers automatically by the ICAP Server iApp– is, however, not in the scope of this document.

<table>
<thead>
<tr>
<th>Description</th>
<th>Screenshot</th>
</tr>
</thead>
</table>
| 1 From the left side menu select **Main > Local Traffic > Virtual Servers**.  
The list of virtual servers is displayed.                                                                                                           | ![Screenshot](image) |
| 2 From the list select the virtual server to which content adaptation needs to be configured.  
The properties of the virtual server are displayed.  
Select **Advanced for Configuration**.                                                                                                            | ![Screenshot](image) |
3 Scroll down and look for Request Adapt Profile and Response Adapt Profile.

Select the Metadefender_ICAP_Request and Metadefender_ICAP_Response profiles based on whether both requests and responses, only one, or none of them needs to be sent to the ICAP Server.
4. Scroll further down and click the *Update* button to commit the changes.
Testing the configuration

To test the configuration point a browser to the IP or DNS address of the virtual server and observe how the traffic is handled by the ICAP Server.

<table>
<thead>
<tr>
<th>Description</th>
<th>Screenshot</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 Obtain the IP address of the virtual server from the configuration.</td>
<td></td>
</tr>
<tr>
<td>Description</td>
<td>Screenshot</td>
</tr>
<tr>
<td>-------------</td>
<td>------------</td>
</tr>
<tr>
<td>2</td>
<td>Point a browser to the web pages of the virtual server and generate traffic.</td>
</tr>
<tr>
<td>Description</td>
<td>Screenshot</td>
</tr>
<tr>
<td>-----------------------------------------------------------------</td>
<td>------------</td>
</tr>
<tr>
<td><strong>3</strong> Observe the traffic in MetaDefender ICAP Server.</td>
<td><img src="image" alt="Screenshot" /></td>
</tr>
</tbody>
</table>

**File blocked**

eicar.txt has been blocked by your administrator's policy (reason: Infected - E)

Your administrator is using OPSWAT's Metadefender Core technology to scan downloads with additional ways Metadefender Core can be used by trying the Metadefender Client demo for at MetaDefender.com file scanning service.
### 4.4.4 McAfee Web Gateway

The current documentation is based on McAfee Web Gateway version 7.7.

**Prerequisites**

- McAfee Web Gateway is installed and license is activated
- MetaDefender ICAP Server is started and configured with "**Use persistent connections**" option enabled. For details please check [3. Configuring MetaDefender ICAP Server](#).
Configure McAfee Web Gateway to use MetaDefender ICAP Server

1. In your browser navigate to the McAfee Web Gateway’s user interface. By default it is accessible via http://<IP address>:4711 or https://<IP address>:4712. Default user/password combination is admin/webgateway.

2. Choose Policy.

3. Under the Rule Sets tab select Add → Rule Set from Library...

4. Select ICAP Client → ICAP Client from the rule set list and click OK.
5. Select the newly created **ICAP Client** under **Rule Sets** and click on **Edit...** next to **ReqMod server**.
6. In the **Edit List (ICAP Server)** window under **List content** double-click on the first item. In the new **Edit ICAP Server** window change the URI for the MetaDefender ICAP Server. It should look like `icap://<ICAP IP>:<ICAP port>/OMSScanReq-AV`. Click OK to close the Edit ICAP Server window and click OK again to close the ReqMod server editor window.

![Edit ICAP Server window](image)

7. Repeat steps 5-6 to set the **RespMod server**. The URI for MetaDefender ICAP Server should be `icap://<ICAP IP>:<ICAP port>/OMSScanResp-AV`

8. After everything is configured click **Save Changes** in the top-right corner. McAfee is now configured to use MetaDefender ICAP Server.

**Enabling SSL Scanner**

If you want to inspect contents in HTTPS connections with MetaDefender ICAP Server you should enable **SSL Scanner** in McAfee Web Gateway.

1. In your browser navigate to the McAfee Web Gateway's user interface. By default it is accessible via `http://<IP address>:4711` or `https://<IP address>:4712`. Default user /password combination is `admin/webgateway`

2. Choose **Policy**

3. Under **Rule Sets** select the **SSL Scanner** rule which is disabled by default
4. Check **Enable** option and click **Save Changes**. McAfee is now configured to decrypt HTTPS traffic and send it to MetaDefender ICAP Server unencrypted.

**Troubleshooting**

- To use the McAfee Web Console, you need to enable java in your browser and add the Web Console's url to the trusted sites in Java Config.

- There are notifications or even non-working web pages after enabling SSL Scanner: you should download and install the SSL certificate used by Web Gateway to your browser. You can get the certificate under **Policy → Settings → SSL Client Context with CA → Default CA**. Click **Export...** next to **Certificate Authority** and import the created file to your browser's trusted root certificates.

- If you see "**16000 - NoIcapServerAvailable**" errors: MetaDefender ICAP Server should be configured to use persistent connections. Please check **3. Configuring MetaDefender ICAP Server**.

**4.4.5 Squid**

**Basic Configuration**

Squid configuration should be done by modifying "squid.conf" (e.g., /etc/squid3/squid.conf). Below is an example of a simplified version of configuration for Squid. For more detailed documentation, please refer to the Squid manual.

1. **Enable acl localnet.**

Search for "acl localnet" section, uncomment all "acl localnet" lines. Below is an example of how the configuration might look:

```plaintext
acl localnet src 10.0.0.0/8
```
### ACL Rules

- `acl localnet src 172.16.0.0/12`  # RFC1918 possible internal network
- `acl localnet src 192.168.0.0/16`  # RFC1918 possible internal network
- `acl localnet src fc00::/7`  # RFC 4193 local private network range
- `acl localnet src fe80::/10`  # RFC 4291 link-local (directly plugged) machines

- `acl SSL_ports port 443`  # http
- `acl Safe_ports port 80`  # http
- `acl Safe_ports port 21`  # ftp
- `acl Safe_ports port 443`  # https
- `acl Safe_ports port 70`  # gopher
- `acl Safe_ports port 210`  # wais
- `acl Safe_ports port 1025-65535`  # unregistered ports
- `acl Safe_ports port 280`  # http-mgmt
- `acl Safe_ports port 488`  # gss-http
- `acl Safe_ports port 591`  # filemaker
- `acl Safe_ports port 777`  # multiling http
- `acl CONNECT method CONNECT`

### Configuration

2. Allow localnet and localhost access by adding the following lines. (under the "Recommended minimum Access Permission configuration" section)

```
http_port 3128
http_access allow localnet
http_access allow localhost
http_access deny all
```

3. Enable ICAP and set the Preview Size to 0. (these values don't exist, so you simply add them anywhere in the file)

```
icap_enable on
icap_send_client_ip on
icap_preview_enable on
icap_preview_size 0
icap_service_failure_limit -1
```
Enable ReqMod (upload mode) again, these don't exist, so add them anywhere

```
icap_service metascan_req reqmod_precache bypass=0
icap://<MetaDefender ICAP Server>:1344/OMSScanReq-AV
adaptation_access metascan_req allow all
```

Enable RespMod (download mode) again, these don't exist, so add them anywhere

```
icap_service metascan_resp respmod_precache bypass=0
icap://<MetaDefender ICAP Server>:1344/OMSScanResp-AV
adaptation_access metascan_resp allow all
```

4. Persistent connections

By default, Squid is using persistent connections to the MetaDefender ICAP Server, this feature can be controlled explicitly by the following directive:

```
icap_persistent_connections on/off
```

If persistent connections are enabled in Squid, the same setting should be applied to the ICAP side or Squid might report some ICAP errors. See 3. Configuring MetaDefender ICAP Server.

5. Restart Squid to apply the new configuration.

⚠️ If you are using Squid 3.1 there is a bug in Squid that drops the connection to the ICAP server. You will notice messages in Squid's cache.log file similar to the following:

```
essential ICAP service is down after an options fetch failure: icap://<MetaDefender ICAP Server>:1344/OMSScanReq-AV [down,!opt]
```

This may cause you to get Squid error messages when trying to access websites.
To fix this, either upgrade to Squid 3.2 or higher, or add `icap_persistent_connections off` to your `squid.conf` file.

Scanning HTTPS content

Learn how to configure Squid to scan HTTPS content below. This allows Squid to send HTTPS content to the MetaDefender ICAP Server for scanning purposes.

Requirements

- Version: Tested with 3.5.19
- Squid has to be compiled with SSL support. For further information, please see: [http://docs.diladele.com/](http://docs.diladele.com/)

Configuration

Tell Squid to listen on the following ports by modifying the line

```
http_port 3128
```

to this:

```
http_port 3128 ssl-bump generate-host-certificates=on
dynamic_cert_mem_cache_size=4MB cert=\etc\ssl\myc.pem
```

In addition, the following lines have to be inserted:

```
sslcrtd_program \lib\squid\ssl_crtd.exe -s
\lib\squid\ssl_crtd.exe -s
\var\cache\squid_ssldb -M 4MB
ssl_bump stare all
ssl_bump bump all
```

Certificate

Generate a new root certificate for Squid:

```
openssl.exe req -new -newkey rsa:1024 -days 1000 -nodes -x509 -
keyout myc.pem -out myc.pem
```
After generating a new certificate, the certificate storage should be reinitialized by deleting the `<SQUIDFOLDER>\var\cache\squid_ssldb` folder and running:

```
<SQUIDFOLDER>\lib\squid\ssl_crtd.exe -c -s
<SQUIDFOLDER>\var\cache\squid_ssldb
```

The certificate has to be installed as a root certificate in the browser. Squid has to be restarted after the changes.

### 4.4.6 FortiGate

This documentation is based on FortiGate 5.4.3 documentation using web-based manager and CLI. For different version of FortiGate or missing information, refer to FortiGate user guides.

**Configuration via Web-Based Manager**

**Enabling ICAP feature**

ICAP does not appear by default in the web-based manager, it has to be enabled by going to **System → Feature Select** and enabling **ICAP**. You may need to refresh the page in order to see the changes.

**Adding ICAP server and profile**

1. Add ICAP server
   a. Select **Security Profiles → ICAP Servers**.
   b. Create new or edit an existing entry.
i. The IP address and port should be the ones used by MetaDefender ICAP Server.

2. Add ICAP profile
   
   a. Select **Security Profiles** → **ICAP**.
   
   b. Create new or edit an existing entry.
   
   c. Select the **Enable Request Processing** check-box, select your server’s name from the drop-down list and in the **Path** field, type in ‘OMSScanReq-AV’.
   
   d. Select the **Enable Response Processing** check-box, select your server’s name from the drop-down list and in the **Path** field, type in ‘OMSScanResp-AV’.
   
   e. Apply the changes.
3. Apply the ICAP profile in your policy/policies
   b. Create new or edit an existing policy
   c. In the section 'Security Profiles', switch on ICAP and select the previously created profile.

Configuration via CLI

1. Add ICAP server

Add ICAP server via CLI

```plaintext
config icap server
  edit <icap_server_name>
    set ip-version {4 | 6} [default: 4]
    set ip-address <server_ipv4>
    set ip6-address <server_ipv6>
    set max-connections <int> [default: 100]
    set port <port_int> [default: 1344]
  end
```

The example shown above in the web-based manager would look like this in the CLI:

Add concrete ICAP server via CLI

```plaintext
config icap server
  edit my_icap_server
    set ip-address 172.16.201.36
  end
```

2. Add ICAP profile

Add ICAP profile via CLI

```plaintext
config icap profile
  edit <icap_profile_name>
    set replacemsg-group <grp_name>
```
set request (enable | disable)  
[The following commands are enabled if request is set to 'enable']
  set request-failure {error | bypass}  
  set request-path <path>  
  set request-server <icap_server>
set request-path <path>  
set request-server <icap_server>
set response (enable | disable)  
[The following commands are enabled if response is set to 'enable']
  set response-failure {error | bypass}  
  set response-path <path>
  set response-server <icap_server>
set streaming-content-bypass (enable | disable)
end

The example shown above in the web-based manager would look like this in the CLI:

Add concrate ICAP profile via CLI

config icap profile
  edit my_icap_profile
    set request enable
    set request-failure error
    set request-path OMSScanReq-AV
    set request-server my_icap_server
    set response enable
    set response-failure error
    set response-path OMSScanResp-AV
    set response-server my_icap_server
    set streaming-content-bypass disable
  end

3. Apply the ICAP profile policy (policy can be replaced by explicit-proxy-policy when setting ICAP for an Explicit Proxy Policy)

Apply ICAP profile via CLI

config firewall policy [or policy6 if IPv6 is used]
  edit <index_int>
    set icap-profile <icap_profile_name>
  end

The example shown above in the web-based manager would look like this in the CLI:

Apply concrate ICAP profile via CLI
If you want to disable the ICAP profile for a given policy you should write

```
config firewall policy
   edit 1
       set icap-profile my_icap_profile
   end
```

**Disable ICAP profile via CLI**

```
config firewall policy [or policy6 if IPv6 is used]
   edit <index_int>
       set icap-profile ""
   end
```

**Scanning HTTPS content**

To be able to inspect and scan SSL/SSH traffic you have to enable it in Fortigate. After enabling this option you should download the certificate used by Fortigate and install/import it to the browsers which communicate with Fortigate. Otherwise you might see SSL/security related notifications or errors, or even not working web pages.

The certificate can be downloaded under **Security Profiles → SSL/SSH Inspection**

**Enable SSL inception via GUI**

1. Navigate to **Policy&Objects** and select the policy you would like to enable SSL inception to (for example an Explicit Proxy Policy)
2. In the Edit page of the selected policy locate the Security Profiles section

3. Turn SSL/SSH Inspection on and set it to deep-inspection

**Enable SSL inspection via CLI**

The steps shown above would like this via CLI:

```
Apply deep ssl inspection to an Explicit Proxy Policy

config firewall explicit-proxy-policy
default edit 1
set ssl-ssh-profile deep-inspection
default edit 1
set ssl-ssh-profile deep-inspection
end
```

**Resources**

- Other FortiGate documentation
- FortiOS Handbook - The Complete Guide to FortiOS
- FortiOS 5.4.1 CLI Reference

**4.4.7 A10 SSL Insight (SSLi)**

- System Requirements
- Configuring MetaDefender ICAP Server
  - Note on MetaDefender ICAP Server Licensing
• Configuring A10 Thunder SSLi

• Testing the configuration

Using A10’s Thunder SSLi together with MetaDefender ICAP Server will help you scan your HTTP and decrypt SSL/TLS traffic flows for threats using all of the engines in MetaDefender Core. This guide describes the basic steps to get MetaDefender ICAP Server working with your A10 Thunder SSLi.

System Requirements
The following systems are required to set up MetaDefender ICAP Server with an A10 Thunder SSLi

• A10 SSL Insight appliance (Thunder SSLi or Thunder CFW)
• MetaDefender ICAP Server
• MetaDefender Core

Configuring MetaDefender ICAP Server
For installation and configuration quick guidelines see: 1. Quick Start with MetaDefender ICAP Server.

For detailed instructions see: 2. Installing or Upgrading MetaDefender ICAP Server and: 3. Configuring MetaDefender ICAP Server.
Note on MetaDefender ICAP Server Licensing

MetaDefender ICAP Server must have a valid license to function correctly. For license configuration details see 2.4. MetaDefender ICAP Server Licensing.

Configuring A10 Thunder SSLi

The steps below describe the minimum configuration required for MetaDefender ICAP Server integration with A10 Thunder SSLi and was created based on ACOS 4.1.4_GR1-P2.

1. Open a web browser and launch the Thunder SSLi ACOS Web GUI. (Please refer to the A10 documentation for details on how to open the ACOS Web GUI. This guide was written with a presumption that you have already completed the device management configuration on the Thunder SSLi.)

2. Navigate to **System > App Template**

3. Select **SSL Insight** to launch the SSLi App Centric Template (ACT)
4. Navigate to the first tab, **Wizard** on the **SSL Insight ACT** (if not automatically directed to it).

5. Select **Custom** under the **Deployment** sub-tab, since the topology would require advanced configuration that needs an IP address to be assigned on the Thunder SSLi appliance. Click **NEXT** when done.
6. Under **Network**, Assign the device **IP Address** and **Default Gateway**. Click **NEXT** when done.

![Network Configuration](image)

7. Import and assign the **SSL Cert. and Key**. Click **NEXT** when done.

![SSL Cert. and Key](image)

8. Select traffic that needs to be bypassed from SSL inspection (for compliance), for e.g. Finance and Healthcare. Click **NEXT** when done.

![SSL Inspection](image)

9. Review your configuration and click **FINISH** when done.
10. Select APPLY.

11. The configuration will take a few seconds and a pop-up will indicate if the configuration was successful.

12. Navigate to the Configuration tab on the SSL Insight ACT, to set up OPSWAT security features.

13. Select Edit Services icon next to Configured Services on the right-hand side.
14. Under **Edit Services**, find and select **ICAP Based DLP / AV**.

15. Check **Enable ICAP**, then clarify the MD ICAP Server IP Address.

16. The **ICAP Operation** contains a drop down menu. The MD ICAP server can do **REQMOD**, **RESPMOD**, or both.
17. Make sure to add, OMSScanReq-A or OMSScanResp-A, respectively on the URL on the Request Service URI.

18. Select **Done** on the top right.

19. Finally, bind the ICAP policy to either the Decryption or Re-encryption Rule.
20. Select the Edit Rule

![Add a New Rule]

21. Under the Configured Services in the drop-down options, chose ICAP Based DLP / AV.

![Configured Services]
22. Select **Deploy** on the to right.

23. Review and select **APPLY**.

24. Configurations will take a few seconds and a popup will indicate if the configuration was successful.

**Testing the configuration**

To check that your configuration is working as expected try to download an Eicar test file over HTTPS [here](#). If everything was set up properly you should see a blocking page similar to this:
4.5 Other Integration

Next Generation Firewalls:

A Next-Generation Firewall (NGFW) is a hardware or a software based network security platform that is a part of the third generation of firewall technology. NGFWs can prevent sophisticated attacks by enforcing security policies at the application level as well as at the port and protocol levels. NGFWs typically combine a traditional firewall with other network device filtering functionalities, such as an application firewall using in-line deep packet inspection (DPI), and an intrusion prevention system (IPS). Most modern day NGFWs include a proxy with the ability to perform HTTPS decryption and to forward traffic through an Internet Content Adaptation Protocol (ICAP) connection to an external server or device for inspection (typically on a specified ICAP port). This allows administrators to take some of the strain off of the NGFW and free up resources, enabling the NGFW to focus on on things that only it can do.

MetaDefender ICAP Server leverages ICAP to offload specific Internet based content from the firewall and focus on preventing known and unknown threats from entering your corporate network.

⚠️ Any firewall that supports ICAP can be integrated with MetaDefender Core via MetaDefender ICAP Server, since ICAP is a generic wrapper protocol to process HTTP traffic.
Device or appliance specific implementation will, however, most likely require extensive testing before use in production.

By combining a next generation firewall with MetaDefender ICAP Server, organizations can benefit from a consolidated and integrated platform that can protect their network traffic, while ensuring that the appropriate information security policy is applied on both incoming and outgoing web traffic.

4.6 HTTP Header Handling

In this page, we document HTTP header that MetaDefender ICAP Server is using for logging and scanning.

Filename

Flowchart
Details

The steps below detail how MetaDefender ICAP Server parses HTTP information from content-disposition.

1. Check if filename* parameter is present
   a. Get the value of the parameter
   b. Look for the last " " character in the value (see Grammar of filename*'s value)
      i. If no " " character is present then the format considered to be invalid (no filename or extension is returned)
      ii. If " " is found then the text after it is considered as the filename (not decoded)
   c. Extension is considered to be the substring after the last "." in the filename

2. If filename* is not present then look for filename
   a. Get the value of the parameter
   b. Check if the first char is a " " (quote mark)
      i. If we start with a quote mark then the filename will be the substring between the opening and closing quote mark (if no closing quote mark is present we consider it as an invalid and no filename or extension is returned)
      ii. Otherwise the whole value is considered as the filename
   c. Extension is considered to be the substring after the last "." in the filename

3. If none of the parameters are present, then give it a try with name
   a. If name parameter is present then the filename will be the parameter's value (between quote mark) and the extension will be empty

4. If none of the above succeeded or the extension contains special characters then the filename and extension parsing is failed in this step.

Good examples

The following table includes good Content-Disposition examples.

<table>
<thead>
<tr>
<th>Parameter format example</th>
<th>Filename</th>
<th>Extension</th>
</tr>
</thead>
<tbody>
<tr>
<td>filename=&quot;utf8&quot;test%20test.a.b.c</td>
<td>test%20test.a.b.c</td>
<td>c</td>
</tr>
<tr>
<td>filename=&quot;test.txt&quot;</td>
<td>test.txt</td>
<td>txt</td>
</tr>
</tbody>
</table>
Parameter format example | Filename | Extension |
---|---|---|
filename=test.txt | test.txt | txt |
filename="test a.txt"; filename*=utf8'en'test%20a.txt | test%20a.txt | txt |
filename = test.txt | test.txt | txt |
name="test" | test |

**Bad examples**
The following table includes bad Content-Disposition examples.

| Parameter format examle | Error |
---|---|
filename="utf8test.txt" | No " " char |
filename="test.txt" | No closing quote mark |
filename=test.t?x | Special character in extension |
name=test | No quote marks |

**Parse from URL**
Parsing is made using the URL via the GET HTTP method. This method receives the text after the last "/" in the URL path until the first "?" character or until the first "#" character if "?" is not present, or until the end of the URL is neither is present.

The extension is considered to be the substring after the last "." character. If the extension is empty or contains special characters, it is considered a failure.

**Good examples**
The following table includes good URL examples.

| URL | Filename | Extension |
---|---|---|
/a/b/c/test.txt | test.txt | txt |
4.7.3

<table>
<thead>
<tr>
<th>URL</th>
<th>Filename</th>
<th>Extension</th>
</tr>
</thead>
<tbody>
<tr>
<td>/test.a.b</td>
<td>test.a.b</td>
<td>b</td>
</tr>
<tr>
<td>/a/test.php?get=x.txt</td>
<td>test.php</td>
<td>php</td>
</tr>
<tr>
<td>/a/test.php#abc</td>
<td>test.php</td>
<td>php</td>
</tr>
<tr>
<td>/a/b/test.php?abc&amp;d=b&amp;g#hjk</td>
<td>test.php</td>
<td>php</td>
</tr>
</tbody>
</table>

### Bad examples

The following table includes bad URL examples.

<table>
<thead>
<tr>
<th>URL</th>
<th>Error</th>
</tr>
</thead>
<tbody>
<tr>
<td>/test</td>
<td>No extension</td>
</tr>
<tr>
<td>/test/?</td>
<td>Empty part between the last &quot;/&quot; and &quot;?&quot;</td>
</tr>
</tbody>
</table>

### Additional Details

The following parameters are handled case insensitively: filename, filename*. It means that FILENAME, FiLeNaMe* etc. will be also found.

Optional whitespaces are handled in the Content-Disposition header. So both "filename=test.txt" and "filename = test.txt" is supported now.

### 4.7 Data trickling

⚠️ **Risk of potential threats allowed**

**WARNING!**

Given by the nature of the thing, enabling data trickling may cause the ICAP Server to release potentially malicious content.

Always enable data trickling with care; only if you know what you are doing, if you are aware of the risk, and if you accept this risk.
The problem

The Internet Content Adaption Protocol (ICAP) allows ICAP clients to pass HTTP messages to ICAP servers for some sort of transformation or other processing ("adaptation"). MetaDefender ICAP Server provides ICAP interface on top of MetaDefender Core. When a user uploads data over HTTP (for example with a PUT or POST request), the contents of the request are forwarded to MetaDefender Core by MetaDefender ICAP Server for scanning. When a user downloads data from an external server (for example with a GET request), the contents of the reply are also forwarded for scanning, before being sent to the user’s computer.

To be able to scan an HTTP request or response, the ICAP Server must have the whole content. It means that the ICAP Server must wait while all the contents arrive before it can start the scanning.

Moreover, before the ICAP server can have a verdict (whether the contents of the request or response are clean or infected), it must wait for the scan to complete.

Both to wait for the contents to arrive and to perform the scan may take for a long time, especially in case of huge files. As the request or response is blocked in the meantime, the end-user may have a bad downloading experience: she may think that the download stalled, or in extreme cases her download may even time out.

Solution: data trickling

Data Trickling is designed to prevent the timeouts that can sometimes be associated with patience pages. To prevent such timeouts, data trickling trickles—or transmits at a very slow rate—bytes of the original contents—we call them drips—to the client at the beginning of the scan. Because the ICAP Server begins serving content without waiting for the scan result, timeouts do not occur. However, to maintain security, delivery of the full object can be withheld until the results of the content scan are complete (and the object is determined to not be infected).

The end-user will be able to download the already released drips. This can help to
1. Give a better downloading experience: the user will see download progress even while the ICAP Server is still processing;

2. Keep the browser session alive for huge files where the scanning may take long (even causing the download to time out).

Data trickling may start after a pre-configured delay, and with pre-configured drip size. As previously mentioned, it is also possible, to withhold the last portion of the contents for security considerations. For further details about configuring data trickling, see the Data trickling section in 4.2 Security rules.

⚠️ **Trickling prefers usability to security**

Data trickling is a usability feature that sacrifices security to provide better user experience in downloading, and to prevent long scans to cause timeouts.

### How it works

For details about configuring data trickling, see the Data trickling section in 4.2 Security rules.

1. The proxy forwards the contents of the request or the response to the ICAP Server.

2. The ICAP Server submits the contents to the Core. If trickling is enabled, then sets its timer to the FIRTS DRIP / DELAY.
   a. If the response from Core arrives before the timer elapses, then
      i. If the contents were allowed, then ICAP Server releases the contents that were returned by Core (see Limitations and drawbacks).
      ii. If the contents were blocked, then the ICAP Server returns the block page (see 3.8 Customizing the block page).
   b. If the response from Core does not arrive before the time elapses then
      i. If ENABLE TO WITHHOLD then
         1. If ENABLE TO WITHHOLD / SIZE would still remain after the drip, then
            a. The first drip is released in the pre-configured size;
            b. The timer is set to the ADDITIONAL DRIPS / DELAY.
         2. Else no drip is released, ICAP Server waits for the scan to complete.
      ii. Else the drip is released
      c. If the response from Core does not arrive before the time elapses then
i. If ENABLE TO WITHHOLD then
   1. If ENABLE TO WITHHOLD / SIZE would still remain after the drip, then
      a. The additional drop is released in the pre-configured size;
      b. The timer is set to the ADDITIONAL DRIPS / DELAY.
   2. Else no drip is released, ICAP Server waits for the scan to complete.

ii. Else the drip is released

⚠️ Risk of releasing malicious object

If ENABLE TO WITHHOLD is not set, and the end of the file is reached any time during the trickling, then the whole original, potentially malicious content has been already released.

Always set ENABLE TO WITHHOLD, except only if you know what you are doing, if you are aware of the risk, and if you accept this risk.

e. If the scan completes, then
   i. If the contents were allowed, then ICAP Server releases the contents that were returned by Core (see Limitations and drawbacks).
   ii. If the contents were blocked, then ICAP Server aborts the connection.

⚠️ No cancel

WARNING!
After some drips have already been released, there is now way to cancel or call them back.

Error rendering macro 'drawio' : null

Limitations and drawbacks

Security

With data trickling enabled, the ICAP Server releases drips of the original contents, before the processing — e.g. anti-malware scanning— completes. These drips being part of the original contents, may contain malicious code.
As a consequence, with data trickling enabled, the risk—that malicious content reaches the end-user or the upload server—is much higher.

⚠️ Risk of potential threats allowed

WARNING!

Given by the nature of the thing, enabling data trickling may cause the ICAP Server to release potentially malicious content.

Always enable data trickling with care; only if you know what you are doing, if you are aware of the risk, and if you accept this risk.

Partial content

⚠️ Risk of potential threats allowed

All major browsers are capable to handle partial downloads and support to automatically retry previously failed, incomplete downloads. In these cases it is very common to request only the missing part of the content using range requests. As with trickling ICAP Server releases drips of the original content, the risk is even higher that range requests will be in use and malicious content may finally be downloaded.

For further details see: 4.8 Risks with range request.

Block page

⚠️ No cancel

WARNING!

After some drips have already been released, there is now way to cancel or call them back.

After the trickling has already been started, there is no point in serving the block page. The reason for that is that the block page would be broken caused by the already released original content.

Deep CDR and Proactive DLP

⚠️ No cancel
WARNING!

After some drips have already been released, there is now way to cancel or call them back.

After the trickling has already been started, there is no point in serving contents that have been processed by Core (e.g. sanitized files by Deep CDR or processed files by Proactive DLP). The reason for that is that the processed contents would be broken caused by the already released original content.

✔ Configure first drip accordingly

A potential resolution may be to choose a FIRST DRIP / DELAY value carefully so that reasonable contents can still finish processing before the first drip is released.

✔ Configure rule filters accordingly

An other potential solution may be to apply trickling to certain ICAP Server rules only. Applying filters to rules, you can select content types (e.g. configuring the Content-Type header as a filter), where data trickling may not be allowed, thus making content processing (CDR, redaction, metadata removal, etc.) usable.

Example

The following filter will match PDF files (by the content type). Disabling data trickling for this rule can make Deep CDR and/or Proactive DLP usable for PDF files.

4.8 Risks with range request

Hypertext Transfer Protocol (HTTP) clients often encounter interrupted data transfers as a result of canceled requests or dropped connections. When a client has stored a partial representation, it is desirable to request the remainder of that representation in a subsequent request rather than transfer the entire representation. Likewise, devices with limited local storage might benefit from being able to request only a subset of a larger representation, such as a single page of a very large document, or the dimensions of an embedded image. [RFC 7233]

Both clients and servers support requesting or sending only parts of the original content based on the request of the client. The fact that a content was sent in fragments is hidden form the end user in most of the cases, it may happen automated by the client and the server to optimize performance.

In most of the cases MetaDefender ICAP Server requires, however, the whole object, so that it can detect malicious contents.
When the content is delivered utilizing range requests, then there is no way for the ICAP Server to have the object as a whole. As a consequence in these cases ICAP Server may not be able to detect existing malicious code in the whole object, as it is assembled at the client side only.

Error rendering macro 'drawio' : null

**Details**

The reason why ICAP needs the whole object is, that certain malware detection methods are based on signatures (similar to hashing), that may be completely different, when only part of the malicious code is available.

As the ICAP protocol is request / response based (similarly to REST) it is also not possible for the ICAP Server to wait for all fragments of the whole object and assemble it for scanning on the ICAP Server side.

**Risk of potential threats allowed**

As a consequence, when range requests are in use, chopped malicious contents may remain undetected by the ICAP Server, and get allowed.

When the client reassembles the original object from the fragments, the resulted contents may be malicious.
5. Troubleshooting MetaDefender ICAP Server

- 5.1 How to create support package
- 5.2 How to read the MetaDefender ICAP Server log
- 5.3 Inaccessible Web Management Console
- 5.4 Startup failure
- 5.5 Can't view scan details

5.1 How to create support package

ℹ️ A support package contains essential information regarding the operating system and OPSWAT software found on the machine.

⚠️ Security and privacy concerns

The contents of the generated package may contain sensitive information that may require special handling.

Linux

To create a package you must start the script found under /usr/bin/mdicapsrv-collect-support-data.sh.

As the script processes the necessary information, the script generates the support package output.

The package files is a tar.gz archive with the following name:

```
mdicapsrv-<VERSION>-support-<TIMESTAMP>.tar.gz
```

Where the timestamp is the date when the package was generated.

Example:

```
mdicapsrv-4.2.0-support-1439983514.tar.gz
```
The generated package will be placed in the same location as the script that was called.

**Windows**

To create a package you must start the script found under the installation directory of the product, default this is `C:\Program Files\Metadefender ICAP Server\mdicapsrv-collect-support-data.bat`.

As the script processes the necessary information, the script generates the support package output.

The package files is a zip archive with the following name:

```
mdicapsrv-<VERSION>-support-<TIMESTAMP>.zip
```

Where the timestamp is the date when the package was generated.

Example:

```
mdicapsrv-4.2.0-support-1439983514.zip
```

The generated package will be placed in the same location as the script that was called.

**Content of the created package**

The support package contains the following elements:

<table>
<thead>
<tr>
<th>Relative path</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>config</td>
<td>directory</td>
<td>The configuration files of OPSWAT software found on the machine</td>
</tr>
<tr>
<td>config/config. copy</td>
<td>file (binary)</td>
<td>Config database <strong>without</strong> user data</td>
</tr>
<tr>
<td>log</td>
<td>directory</td>
<td>The log files of OPSWAT software found on the machine</td>
</tr>
<tr>
<td>os.info</td>
<td>file (text)</td>
<td>Operating system information</td>
</tr>
<tr>
<td>hw.info</td>
<td>file (text)</td>
<td>Hardware information</td>
</tr>
<tr>
<td></td>
<td>file (text)</td>
<td>Network information</td>
</tr>
</tbody>
</table>
### Relative path

<table>
<thead>
<tr>
<th>Relative path</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>network.info</td>
<td></td>
<td></td>
</tr>
<tr>
<td>files.info</td>
<td>file (text)</td>
<td>OPSWAT software directory information</td>
</tr>
</tbody>
</table>

⚠️ You are recommended to check the content of the generated package to make sure it does not contain any confidential information.

## 5.2 How to read the MetaDefender ICAP Server log

- **Linux**
  - Files
  - Format
  - Severity levels of log entries
- **Windows**

### Linux

The log files are plain text files that can be opened with any text editor.

### Files

The MetaDefender ICAP Server generates a log file under `/var/log/mdicapsrv` named `mdicapsrv.log`.

### Format

In the log, each line represents a log message sent by the server. Depending on the log file, the format of the line is as follows:

```
[LEVEL] TIMESTAMP (COMPONENT) MESSAGE [msgid: MESSAGE ID]
```

Example:
Where the different values are:

- **LEVEL**: the severity of the message
- **TIMESTAMP**: The date value when the log entry was sent
- **COMPONENT**: which component sent the entry
- **MESSAGE**: the verbose string of the entry's message
- **MESSAGE ID**: the unique ID of this log entry

**Severity levels of log entries**

Depending on the reason for the log entry, there are different types of severity levels. Based on the configuration, the following levels are possible:

- **DUMP**: The most verbose severity level, these entries are for debuggers only.
- **DEBUG**: Debuggers severity level, mostly used by support issues.
- **INFO**: Information from the software, such as scan results.
- **WARNING**: A problem occurred needs investigation and OPSWAT support must be contacted, however the product is supposed to be operational.
- **ERROR**: Software error happened, please contact support if the issue is persist. Software functionality may be downgraded in these cases.

**Windows**

On Windows systems logging is done via Event Log.

MetaDefender ICAP Server logs can be found under **Windows Logs / Application** and are labelled with **Metadefender ICAP** source.

### 5.3 Inaccessible Web Management Console

**Connection refused or browser is waiting**

The Web Management Console can not be accessed from browser. You get an error message (connection refused) or your browser is waiting for reply.
Solution

1. Please make sure your computer can access the MetaDefender ICAP Server IP address.
2. Please make sure you entered the correct URL into your browser.
3. Please make sure you opened the firewall port on the MetaDefender ICAP Server server for the Web Management Console. Consult your Linux Distribution manual on how to configure a firewall in your distribution.

Invalid license

The Web Management Console can not be accessed from browser. You get the following message:

File blocked
End-user client's license identifier is missing

Your administrator is using OPSWAT's Metadefender Core technology to scan downloads with multiple anti-malware engines. See additional ways Metadefender Core can be used by trying the Metadefender Client demo for scanning endpoint processes and files or the Metadefender.com file scanning service.

The problem is that:

- The MetaDefender ICAP Server license is not valid (not licensed at all or maximum client number is reached)
- The Web Management Console is accessed via a Proxy server or other ICAP client
Solution
Try accessing the Web Management Console directly from the machine that hosts the ICAP Server or with no Proxy connection.

5.4 Startup failure

Could not create temporary directory

MetaDefender ICAP Server requires temporary directories and files for its operation. If the temporary directory can not be created then the ICAP Server will fail to start with *Could not create temporary directory* message in the logs.

Example log message on CentOS 7:

```
```

Consider the following:

1. Proper access rights are configured on the temporary directory root (see the `tempdirectory` entry in 3.1.2 MetaDefender ICAP Server configuration file)
2. There is appropriate amount of free disk space for temporary files on the partition where temporary directory root is configured to
   a. In case of abnormal termination of ICAP Server (crash, power supply problems etc.) there is a chance that temporary files remain under the temporary directory root.

   ! These remaining temporary files require manual cleanup.

5.5 Can't view scan details

The View scan details link on Dashboard > ICAP history > Request details points to the scan details on the MetaDefender Core instance where the actual scanning took place.

For further details see 4.1. Dashboard.
Symptoms

Dashboard instead of scan details
The View scan details link leads to the Dashboard on the referenced Core instead of the scan details.

Reason
The View scan details link works with MetaDefender Core version 4.7.0 (or later) only.

Connection refused or browser is waiting
The Web Management Console can not be accessed from the browser. You get an error message (connection refused) or your browser is waiting for reply.

Reason
View scan details links stop working after the referenced Core is uninstalled or migrated to a new address.

Page not found
The browser displays a page not found error, however everything seems to be configured properly.

Reason
View scan details links use the IP addresses configured at the Core inventory. If these IP addresses are not accessible from the machine where the actual browsing happens, then the details page can not be displayed.
6. Release Notes

| 4.7.3  
MetaDefender ICAP Server Release  
3 February, 2020 |
<table>
<thead>
<tr>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>MetaDefender ICAP Server 4.7.3 is a maintenance release providing minor improvements and fixes.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>New &amp; Improved</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Export ICAP History to CSV</strong></td>
</tr>
<tr>
<td>MetaDefender ICAP Server now supports exporting Dashboard &gt; ICAP History to CSV. For further details see the Operations / Export to CSV section in 4.1. Dashboard.</td>
</tr>
</tbody>
</table>
ICAP Server block page placeholder for violations (%%%violations%%% ) can be a quite long list in certain cases as different engines might use different names for the same threat. ICAP Server 4.7.3 introduced the new placeholder (%%%violations_short%%%). For further details see 3.8 Customizing the block page.
4.7.3

Multi-part requests blocked

In case of multi-part HTTP messages (HTML forms, for example) ICAP Server tried to scan even the empty parts (e.g. a file upload input that does not contain a file). In certain cases this resulted in blocking the message even if Core 4.17.0 (and newer) that blocks empty files.

Server profile details hidden

When ICAP Server was managed by OPSWAT Central Management, the user could not even view the server profile details.

⚠️ Settings managed by Central Management

When ICAP Server is managed by Central Management, then settings are read-only on the instance. It means that even after this fix it will not be possible to edit server profile settings, only to view them, when the instance is centrally managed.

Previous releases

<table>
<thead>
<tr>
<th>Feature</th>
<th>Maintenance</th>
</tr>
</thead>
<tbody>
<tr>
<td>4.7.0</td>
<td>4.7.1</td>
</tr>
<tr>
<td>4.6.0</td>
<td>4.6.1</td>
</tr>
<tr>
<td>4.5.0</td>
<td></td>
</tr>
<tr>
<td>4.4.0</td>
<td></td>
</tr>
</tbody>
</table>

6.1 Changelog

- Version 4.7.3
- Version 4.7.2
- Version 4.7.1
- Version 4.7.0
- Version 4.6.1
- Version 4.6.0
- Version 4.5.0
- Version 4.4.0
Version 4.7.3

Enhancement and new features:
- Export ICAP History to CSV
- Placeholder for first violation

Fixed issues:
- Multi-part requests blocked
- Server profile details hidden

Version 4.7.2

Fixed issues:
- Upgrade failure in version 4.7.1

Version 4.7.1

Enhancement and new features:
- Support for product sets and groups in Central Management

Version 4.7.0

Enhancement and new features:
- ICAP Server data trickling
Fixed issues:
- Config wizard loading for ever

Version 4.6.1
Fixed issues:
- There were some broken links in the user guide page F5 BIG IP SSL configuration
- Product schema is invalid error when adding ICAP Server to Central Management

Version 4.6.0
Enhancement and new features:
- Permissive parsing to accept space in header names
- Support additional placeholder tokens in the block page
- Rules to support filtering by the ICAP client (proxy)

Version 4.5.0
Enhancement and new features:
- Added support for ICAP preview functionality
- Permissive parsing now accepts HTTP requests/responses without HTTP version
- Syslog messages can be configured to use the server's local timezone
- Added an option to enforce an enhanced password policy to local user directories
- Number of all and displayed requests are shown in the ICAP history
- Improved history loading and service starting time
Fixed issues:
- Content-Length header was missing from blocked responses
- Fixed some minor issues on the default block page

Version 4.4.0
Enhancement and new features:
- Added additional pages to wizard for easier setup of the product
- Added a more advanced filtering to the history pages next to the basic search
- Welcome wizard now allows creating user with the name "admin"
- Added a bypass option for general Core errors
• Added an option for requests with not supported encoding or decoding error to be scanned as it is

Fixed issues:
• Decreased severity of "An error occurred during writing response" log message from warning to debug
• Security rules could disappear after service restart if a server profile was deleted which was used in an already deleted security rule
• Updated texts and URLs in the default block page's footnote

Version 4.3.1
Fixed issues:
• Improved temporary directory creation as it could fail on Windows when there were 256 directories present at the same time

Version 4.3.0
Important features:
• Introduced welcome wizard
• Batch scanning functionality of MetaDefender Core is used when handling multipart requests

Enhancement and new features:
• Added wildcard (globbing) support for Host and Client IP matching in Security rules
• Added support for Layer 7 load balanced Core
• Added an option for skipping scanning parts in a multipart request where the Content-Disposition header does not contain a filename parameter

Fixed issues:
• Windows event log source was changed to "Metadefender ICAP Server" from the incorrect "Metadefender ICAP"
• Support package didn't remove every temporary file when cleaning up

Version 4.2.3
Fixed issues:
• Results were missing from the ICAP history
Version 4.2.2
New features:
  • Parallelized processing of multipart requests
Fixed issues:
  • Central Management connectivity fix

Version 4.2.1
New features:
  • Re-interpreted, fresh look and feel
  • Central Management connectivity

Version 4.2.0
New features:
  • Support deflate encoding / compression
  • Configurable timeout for request scans
  • Do not send oversized requests to Core
  • Display IPs that are counted for license
  • Search for keywords in ICAP history URIs
  • Display blocked requests on the chart
Fixed issues:
  • Read-only users can't view inventory

Version 4.1.1
Fixed issues:
  • Option to capture traffic of bad requests
  • Option to use more permissive parsing

Version 4.1.0
New features:
  • Core load balancing and high availability
  • Option to override blocking oversized files
• Detailed error messages in block page
• Detailed error messages in ICAP history

**Version 4.0.3**

Fixed issues:
• License option for unlimited number of clients

**Version 4.0.2**

Fixed issues:
• Blocking page is not configurable

**Version 4.0.1**

Fixed issues:
• Upgrade clears all data

**Version 4.0.0**

New features:
• Standalone product offering
• Support for multiple Linux distributions
• Web based user interface for management, configuration and monitoring
• Role based user management with multiple admin users and Active Directory integration
• Multiple ICAP security policies with filters and advanced scan configuration
• Different policy filter settings based on source client, destination host, or any other header
• Multi-part (MIME) sanitization
• Support for base64, brotli and gzip encoding
6.2 Archived release notes
### 6.2.1 Version 4.4.0

<table>
<thead>
<tr>
<th>4.4.0 MetaDefender ICAP Server release</th>
<th>MetaDefender ICAP Server 4.4.0 is a feature release focused on welcome w</th>
</tr>
</thead>
<tbody>
<tr>
<td>26 September, 2018</td>
<td></td>
</tr>
</tbody>
</table>

### New & improved

<table>
<thead>
<tr>
<th>Added additional pages to the Welcome wizard</th>
<th>Now you can also connect to a MetaDefender Core instance and create your wizard.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Welcome wizard now allows creation of the &quot;admin&quot; user</td>
<td>In the first version of the wizard, you were unable to create an account with tl</td>
</tr>
<tr>
<td>Added advanced filtering to the ICAP history pages</td>
<td>Now you can find a more sophisticated way to search for specific ICAP reque</td>
</tr>
<tr>
<td></td>
<td><em>Result, HTTP URI, Client IP and ICAP rule.</em></td>
</tr>
</tbody>
</table>
Added a bypass option for general Core errors

Now you have another override option in Security Rules → Advanced for allowing requests where a general MetaDefender Core server error (with status 500) occurred during the processing.

Added an option for requests with unsupported encoding or decoding errors to be scanned as is

In the past, requests with unsupported encoding (from Content-Encoding header) or requests where a decoding error occurred could have been blocked or bypassed. Now you can select if you would like to process those requests in Security Rules → Advanced.

Fixed

Security rules could disappear after service restart

Due to a configuration validation issue (validating inactive elements), security rules could disappear after a service restart. This would occur if before the service restart you deleted a security rule, and after deletion you also deleted the MetaDefender Core server profile it was using.
<table>
<thead>
<tr>
<th>Updated text and URLs in the default block page's footnote</th>
<th>There were some obsolete URLs and text in the default block page's footnote</th>
</tr>
</thead>
<tbody>
<tr>
<td>Decreased the severity of &quot;An error occurred during writing response&quot; log message from warning to debug</td>
<td>There could be a lot of &quot;An error occurred during writing response&quot; warning because the clients were closing the connection while MetaDefender ICAP S issue.</td>
</tr>
</tbody>
</table>
## 6.2.2 Version 4.5.0

<table>
<thead>
<tr>
<th>4.5.0 MetaDefender ICAP Server release</th>
<th>MetaDefender ICAP Server 4.5.0 is a feature release focused on preview functionality.</th>
</tr>
</thead>
<tbody>
<tr>
<td>19 December, 2018</td>
<td></td>
</tr>
</tbody>
</table>

### New & improved

<table>
<thead>
<tr>
<th>Added support for ICAP preview functionality</th>
<th>Message Preview described in <a href="https://tools.ietf.org/html/rfc3507#page-18">https://tools.ietf.org/html/rfc3507#page-18</a> is now supported.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Permissive parsing now accepts HTTP requests</td>
<td>Previously permissive parsing blocked requests with &quot;Bad request&quot; result where the HTTP version was missing. These requests are now processed.</td>
</tr>
<tr>
<td>responses without HTTP version</td>
<td>Example:</td>
</tr>
<tr>
<td></td>
<td>200 OK</td>
</tr>
<tr>
<td></td>
<td>Content-Length: 58</td>
</tr>
<tr>
<td></td>
<td>Date: Thu, 30 Aug 2018 08:55:33 GMT</td>
</tr>
<tr>
<td></td>
<td>Content-Type: application/octet-stream</td>
</tr>
<tr>
<td></td>
<td>....</td>
</tr>
<tr>
<td>Syslog messages can be configured to use</td>
<td>If one would like to see the syslog messages in the server's timezone instead of UTC, it is now possible.</td>
</tr>
<tr>
<td>the server's local timezone</td>
<td></td>
</tr>
<tr>
<td>Added an option to enforce an enhanced password policy to local user directories</td>
<td>It is now possible to enforce users in local directories to use stronger passwords.</td>
</tr>
<tr>
<td>---</td>
<td>---</td>
</tr>
<tr>
<td>Number of all and displayed requests are shown in the ICAP history</td>
<td><img src="image1.png" alt="Table showing request history" /></td>
</tr>
<tr>
<td>Improved history loading and service starting time</td>
<td>History queries were updated so the ICAP service can start faster and loading the history also takes less time.</td>
</tr>
<tr>
<td>Fixed</td>
<td>Content-Length header was not included in the block response making browsers wait for more data and the page got timeout.</td>
</tr>
<tr>
<td>Fixed some minor issues</td>
<td>Made some small fixes so the default block page is valid HTML5.</td>
</tr>
</tbody>
</table>
on the default block page
## 6.2.3 Version 4.6.0

**4.6.0 MetaDefender ICAP Server release**  
14 June, 2019

### New & improved

<table>
<thead>
<tr>
<th>Permissive parsing to accept space in header names</th>
<th>When <strong>Settings &gt; Global Settings</strong> / ENABLE PERMISSIVE PARSING is enabled, MetaDefender ICAP Server will accept header names that contain a trailing space.</th>
</tr>
</thead>
</table>

#### Example

The following example shows a header that contains a trailing space:

```
Access-Control-Allow-Origin : *
```

By the standards, header names must not contain a trailing space, the colon must follow immediately. Some customers, however, experienced this issue, and received *Invalid character found in HTTP response header name* errors.

<table>
<thead>
<tr>
<th>Support further placeholder tokens in the block page</th>
<th>In customized block pages it is now possible to show information about:</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>• The URL parsed from the original request,</td>
</tr>
<tr>
<td></td>
<td>• Address of the client from whom the request originated,</td>
</tr>
<tr>
<td></td>
<td>• Name of the server the ICAP Server is running on,</td>
</tr>
<tr>
<td></td>
<td>• Date &amp; time of the arrival of the request to the ICAP Server,</td>
</tr>
<tr>
<td></td>
<td>• Names of the files that were blocked,</td>
</tr>
<tr>
<td></td>
<td>• Media types for files that were blocked,</td>
</tr>
<tr>
<td></td>
<td>• Violations for blocked files (e.g. infection, size limit).</td>
</tr>
</tbody>
</table>

⚠️ **Not in the default block page**
These new placeholder tokens are not used in the default block page, but are available for a custom block page. For further details see 3.8 Customizing the block page.

Rules to support filtering by the ICAP client (proxy)

Under Policy > Security Rules / add/modify / REQUEST FILTERS MetaDefender ICAP Server now supports filtering requests by the IP address of the ICAP client (e.g. proxy).

This filter - for example - helps to assign specific rules to traffic that arrives from specific ICAP clients (proxies).
### 6.2.4 Version 4.6.1

<table>
<thead>
<tr>
<th>4.6.1 MetaDefender ICAP Server release</th>
<th>MetaDefender ICAP Server 4.6.1 is a maintenance release focusing on fixing a recent Central Management integration issue.</th>
</tr>
</thead>
<tbody>
<tr>
<td>28 June, 2019</td>
<td></td>
</tr>
</tbody>
</table>

#### Fixed

- **There were some broken links in the user guide page F5 BIG IP SSL configuration**
  - The pages have been moved under the links
    - Managing client-side HTTPS traffic using a self-signed certificate
    - Managing client and server HTTPS traffic using a self-signed certificate
    - Managing client-side HTTPS traffic using a CA-signed certificate
  - on the MetaDefender ICAP Server 4.6.0 / ... / 4.4.3 F5 Integrations / F5 BIG IP SSL configuration page in the user guide.
  - The links now point to the new location of the pages.

- **Product schema is invalid error when adding ICAP Server to Central Management**
  - When trying to add MetaDefender ICAP Server 4.6.0 to the recent versions of Central Management, the operation fails and the *Product schema is invalid* error message is shown.
  - This error is the result of some modifications introduced in ICAP Server 4.6.0, and that is fixed now.
### 6.2.5 Version 4.7.0

<table>
<thead>
<tr>
<th>4.7.0 MetaDefender ICAP Server Release</th>
<th>MetaDefender ICAP Server 4.7.0 is a feature release focusing on the data trickling functionality.</th>
</tr>
</thead>
<tbody>
<tr>
<td>6 August, 2019</td>
<td></td>
</tr>
</tbody>
</table>

#### New & improved

<table>
<thead>
<tr>
<th>ICAP Server Data Trickling</th>
<th>Risk of potential threats allowed</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>![Warning Icon]</td>
</tr>
<tr>
<td></td>
<td><strong>WARNING!</strong></td>
</tr>
<tr>
<td></td>
<td>Given by the nature of the thing, enabling data trickling may cause the ICAP Server to release potentially malicious content.</td>
</tr>
<tr>
<td></td>
<td>Always enable data trickling with care; only if you know what you are doing, if you are aware of the risk, and if you accept this risk.</td>
</tr>
</tbody>
</table>

Data Trickling is designed to prevent timeouts that can sometimes be associated with patience pages. To prevent such timeouts, data trickling trickles—or transmits at a very slow rate—bytes of the **original contents**—we call them drips—to the client at the beginning of the scan. Because the ICAP Server begins serving content without waiting for the scan result, timeouts do not occur. However, to maintain security, delivery of the full object can be withheld until the results of the content scan are complete (and the object is determined to not be infected).

For further details see 4.2 Security rules and 4.7 Data trickling.

#### Fixed

| Config wizard loading for ever | For fresh installations of ICAP Server 4.6.0 and 4.6.1, the basic configuration wizard kept loading endlessly after completing the configuration procedure. |
## 6.2.6 Version 4.7.1

<table>
<thead>
<tr>
<th>4.7.1 MetaDefender ICAP Server Release</th>
<th>MetaDefender ICAP Server 4.7.1 is a maintenance release providing support for Central Management product groups and sets.</th>
</tr>
</thead>
<tbody>
<tr>
<td>26 November, 2019</td>
<td></td>
</tr>
</tbody>
</table>

### New & improved

**Support for product sets and groups in Central Management**

MetaDefender ICAP Server now supports product groups and sets with OPSWAT Central Management. For further details see the section 3.7 Managing OPSWAT Product Sets and Groups in the OPSWAT Central Management user guides.

## 6.2.7 Version 4.7.2

<table>
<thead>
<tr>
<th>4.7.2 MetaDefender ICAP Server Release</th>
<th>MetaDefender ICAP Server 4.7.2 is a maintenance release providing fixing an upgrade bug in the previous version.</th>
</tr>
</thead>
<tbody>
<tr>
<td>09 December, 2019</td>
<td></td>
</tr>
</tbody>
</table>

### Fixed

**Upgrade failure in version 4.7.1**

MetaDefender ICAP Server version 4.7.1 installation failed when there was an attempt to upgrade from a previous version (e.g. from version 4.7.0).
7. Legal

- Copyright
- MetaDefender Export Classification

Copyright

DISCLAIMER OF WARRANTY
OPSWAT Inc. makes no representation or warranties, either express or implied by or with respect to anything in this document, and shall not be liable for any implied warranties of merchantability or fitness for a particular purpose or for any indirect special or consequential damages.

COPYRIGHT NOTICE
OPSWAT, OESIS, Metascan, Metadefender, AppRemover and the OPSWAT logo are trademarks and registered trademarks of OPSWAT, Inc. All other trademarks, trade names and images mentioned and/or used herein belong to their respective owners.

No part of this publication may be reproduced, stored in a retrieval system or transmitted, in any form or by any means (photocopying, recording or otherwise) without prior written consent of OPSWAT Inc. No patent liability is assumed with respect to the use of the information contained herein. While every precaution has been taken in the preparation of this publication, OPSWAT Inc. assumes no responsibility for errors or omissions. This publication and features described herein are subject to change without notice.

MetaDefender Export Classification

MetaDefender United States Export Classification Number (ECCN) is 5D002, subparagraph c.1

Exports and re-exports of MetaDefender are subject to U.S. export controls and sanctions administered by the Commerce Department's Bureau of Industry and Security (BIS) under the U.S. Export Administration Regulations (EAR).

This page provides export control information on MetaDefender. MetaDefender provides encryption features that are subject to the EAR and other U.S. laws. These features have been approved for export from the United States, subject to certain requirements and limitations. You may find the information on this page useful for determining exportability to particular countries or parties, and for completing export or shipping documentation, recordkeeping, or post-shipment reporting.
Although we provide the information on this page, you remain responsible for exporting or re-exporting MetaDefender in accordance with U.S. law. We encourage you to seek appropriate legal advice and/or consult the EAR and the BIS Information Technology Controls Division before exporting, re-exporting, or distributing MetaDefender. The information provided here is subject to change without notice.
How long is the support life cycle for a specific version/release of MetaDefender ICAP Server v4?

OPSWAT provides support on each release of MetaDefender ICAP Server v4 for **18 months** after the publication of the next release of the product (i.e. once a new release is published, you have 18 more months of support on the previous release). However, bug fixes and enhancements are applied only to the next release of a product, not to the current release or historical releases, even when those releases are still under support. In some cases, hot-fixes can be provided for the current release of the product and then incorporated as a regular fix in the next release.

OPSWAT strongly encourages customers to upgrade to the latest release on a regular basis and not to wait until the end of a release supported life-cycle.

<table>
<thead>
<tr>
<th>Release number</th>
<th>Release date</th>
<th>End-of-life date</th>
</tr>
</thead>
<tbody>
<tr>
<td>4.7.2</td>
<td>09 Dec 2019</td>
<td></td>
</tr>
<tr>
<td>4.7.1</td>
<td>26 Nov 2019</td>
<td>09 Jun 2021</td>
</tr>
<tr>
<td>4.7.0</td>
<td>06 Aug 2019</td>
<td>26 May 2021</td>
</tr>
<tr>
<td>4.6.1</td>
<td>27 Jun 2019</td>
<td>06 Feb 2021</td>
</tr>
<tr>
<td>4.6.0</td>
<td>14 Jun 2019</td>
<td>27 Dec 2020</td>
</tr>
<tr>
<td>4.5.0</td>
<td>19 Dec 2018</td>
<td>14 Dec 2020</td>
</tr>
<tr>
<td>4.4.0</td>
<td>26 Sep 2018</td>
<td>19 Jun 2020</td>
</tr>
<tr>
<td>4.3.1</td>
<td>10 Aug 2018</td>
<td>26 Mar 2020</td>
</tr>
</tbody>
</table>
How to upgrade from ICAP Server 4.0.0 to a newer ICAP Server v4 release

MetaDefender ICAP Server v4.0.0 release contains an issue that can cause configuration data loss during the upgrade.

This issue applies to Windows deployments only.
On Linux, the upgrade can be performed on the traditional way, with no data loss.

To upgrade from v4.0.0, please perform the followings:

1. Stop the ICAP Server service using the following command:
2. Check the data folder under MetaDefender ICAP Server installation folder (\<INSTALLATION DIRECTORY>\data, usually C:\Program Files\OPSWAT\MetaDefender ICAP Server\data). If there is any *.war or *.shm files that exist in this folder, it means that the services are still running. Please be sure to stop the services correctly (no mdicapsrv.exe process is running).

3. **Copy** the content of your \<INSTALLATION DIRECTORY>\data folder from the installation folder to a safe place.

4. Uninstall MetaDefender ICAP Server v4.0.0.

5. Remove content of \<INSTALLATION DIRECTORY>\data folder.

6. Install the latest version of the MetaDefender ICAP Server.

7. Stop the ICAP Server service again using the following command:

```
net stop mdicapsrv
```

8. **Replace** the data folder with the files from **Step 3** to \<INSTALLATION DIRECTORY>\data (keeping your data backup untouched).

9. Run the following command: \<INSTALLATION DIRECTORY>\mdicapsrv-upgrade-db.exe.

10. Restart the ICAP Server service using the following command:

```
net start mdicapsrv
```

11. Login to the web management interface and check if your configuration and scan history is untouched.

---

If you need any further help to upgrade from this version, we encourage you to contact [OPSWAT support](#) for assistance.

---

This article applies to MetaDefender ICAP v4

This article was last updated on 2019-10-09
Installing ICAP Server on Debian 9

Symptom
When you install ICAP Server on Debian 9 you may get the following error message:

dpkg: dependency problems prevent configuration of mdicapsrv:
    mdicapsrv depends on libssl1.0.0 (>= 1.0.0); however:
    Package libssl1.0.0 is not installed.

Reason
By default, Debian 9 is installed with libssl1.0.2. For compatibility reasons OPSWAT products require libssl1.0.0. libssl1.0.0 is, however, only available in Debian 8 repositories, and is not available for Debian 9.

Resolution
To install libssl1.0.0 on Debian 9 from a Debian 8 repository perform the following steps:

<table>
<thead>
<tr>
<th>Step</th>
<th>Description</th>
<th>Screenshot / Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Open <a href="https://packages.debian.org/jessie/libssl1.0.0">https://packages.debian.org/jessie/libssl1.0.0</a> in your browser</td>
<td><img src="image" alt="Screenshot" /></td>
</tr>
</tbody>
</table>

Package: libssl1.0.0 (1.0.1t-1+deb8u1)
Secure Sockets Layer toolkit - shared library

This package is part of the OpenSSL project's implementation of cryptographic protocols for secure communication over the Internet. It provides the libssl and libcrypto shared libraries.

Other Packages Related to libssl1.0.0
In the "Downloads" section, at the bottom of the page, select your server's architecture.

<table>
<thead>
<tr>
<th>Architecture</th>
<th>Package Size</th>
<th>Installed Size</th>
</tr>
</thead>
<tbody>
<tr>
<td>amd64</td>
<td>1,024.0 kB</td>
<td>3,080.0 kB</td>
</tr>
<tr>
<td>armel</td>
<td>834.0 kB</td>
<td>2,283.0 kB</td>
</tr>
<tr>
<td>armlphf</td>
<td>850.3 kB</td>
<td>1,865.0 kB</td>
</tr>
<tr>
<td>i386</td>
<td>2,094.9 kB</td>
<td>7,475.0 kB</td>
</tr>
</tbody>
</table>

Select a mirror (http://security.debian.org/debian-security in the example) and download the package.

# wget http://security.debian.org/debian-security/pool/updates/main/o/openssl/libssl1.0.0_1.0.1t-1+deb8u12_amd64.deb

Install the downloaded package.

# dpkg -i libssl1.0.0_1.0.1t-1+deb8u12_amd64.deb

Clean-up the previous broken ICAP Server installation (optional).

# apt --fix-broken install

Re-install ICAP Server.

# dpkg -i mdicapsrv_4.7.0-1_amd64.deb

This article applies to MetaDefender ICAP v4.
This article was last updated on 2019-10-09.