# SSH Communications Security

## CryptoAuditor

# RSA Ready Implementation Guide for RSA Security Analytics

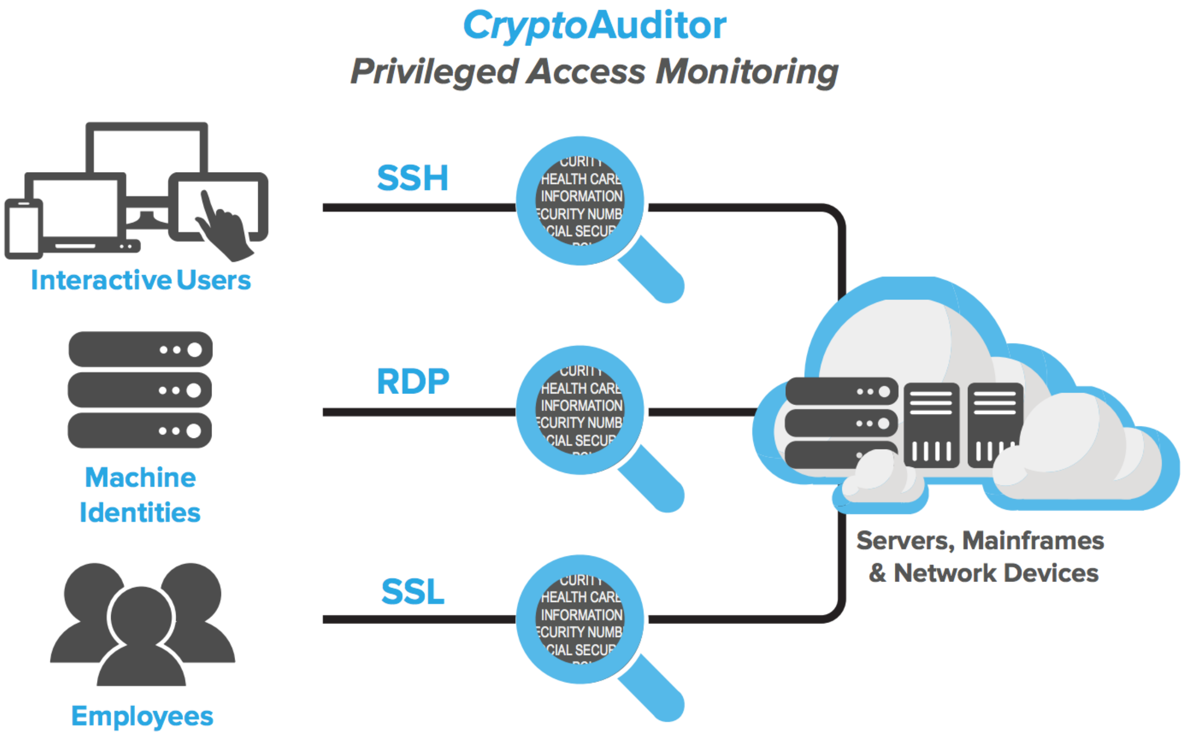
Last Modified: June 16th 2015

## Partner Information

| Product Information | |
|---|---|
| **Partner Name** | SSH Communications Security |
| **Web Site** | www.ssh.com |
| **Product Name** | CryptoAuditor |
| **Version & Platform** | 1.5 Virtual Appliance |
| **Product Description** | Encrypted channels are vital for securing confidential information, but at the same time, they can provide a cloak for malicious activity. CryptoAuditor brings you control and visibility without compromising the security of your encrypted channels. |

## Solution Summary

CryptoAuditor is a transparent and centralized real-time privileged access monitoring and auditing solution that enables organizations to control trusted insider data transfer activities on the fly and without any impact on remote administrators. CryptoAuditor is designed to reduce potential security threats from trusted insiders, meet current and emerging compliance mandates and reduce costs associated with implementation and administration with a minimally invasive approach designed to work with your existing network architecture.  CryptoAuditor works with RSA Security Analytics by sending decrypted SSL traffic to the Security Analytics Packet Decoder for further inspection and analysis.



# Partner Product Configuration

## *Before You Begin*

This section provides instructions for configuring CryptoAuditor for use with RSA Security Analytics. This document is not intended to suggest optimum installations or configurations.

It is assumed that the reader has both working knowledge of all products involved, and the ability to perform the tasks outlined in this section. Administrators should have access to the product documentation for all products in order to install the required components.

All CryptoAuditor components must be installed and working prior to the integration. Perform the necessary tests to confirm that this is true before proceeding.

**! Important: The configuration shown in this Implementation Guide is for example and testing purposes only. It is not intended to be the optimal setup for the device. It is recommended that customers make sure SSH CryptoAuditor is properly configured and secured before deploying to a production environment. For more information, please refer to the CryptoAuditor documentation or website.**

## *CryptoAuditor Configuration*

CryptoAuditor supports HTTP and HTTPS session auditing transparently in router and bridge mode, and through bastion mode Hound's SOCKS proxy listener. HTTP/HTTPS auditing is meaningful for example for auditing admin sessions on web-based admin interfaces, or for auditing file or media transfers over HTTPS.

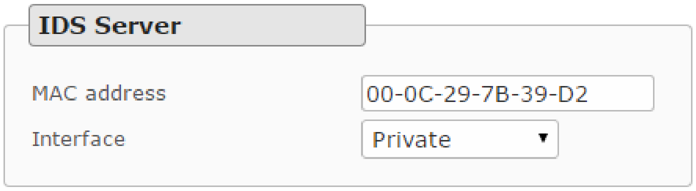
### Setting IDS Server in Hound Configuration

IDS data forwarding is Hound specific, which means that the IDS server must be set for each Hound separately. Then the IDS checkbox must be enabled in the auditing actions of the connection rules that match to the connections of which payload should be passed to the IDS (in this case the SA Packet Decoder).

To set the IDS target server for a Hound, do the following:

1. In the admin UI, navigate to **Settings… Hounds**.
2. In the Hound's **Settings** section, under **IDS Server** section, set IDS settings:

- **MAC address**: MAC address of the SA Packet Decoder network interface. Enter a MAC address prefix that is three octets or more.
- **Interface**: The Hound's source interface that the Hound uses for sending the data to the SA Packet Decoder interface (**private** in this example):



IDS Server

MAC address: 00-0C-29-7B-39-D2

Interface: Private

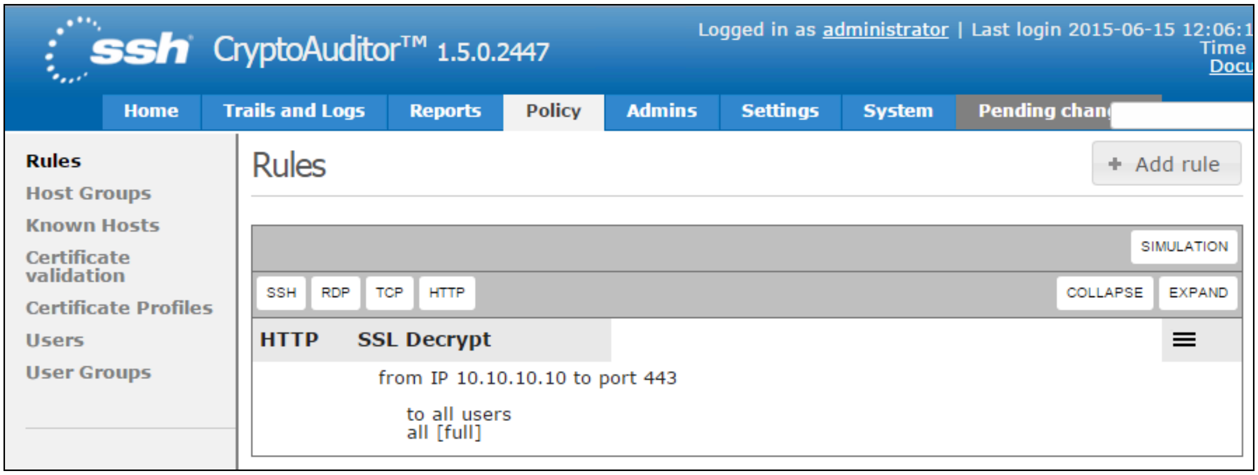
3. To make the changes to take effect in the system, click first **Save**, and then **Apply** the pending changes.

   Now that the IDS interface has been configured, you can create a **Connection Rule** for capturing HTTPS traffic.

## Creating a CryptoAuditor Connection Rule

Before you can audit HTTPS sessions, you must first create a new **Connection Rule**. To create a new Connection Rule for decrypting SSL sessions perform the following steps:

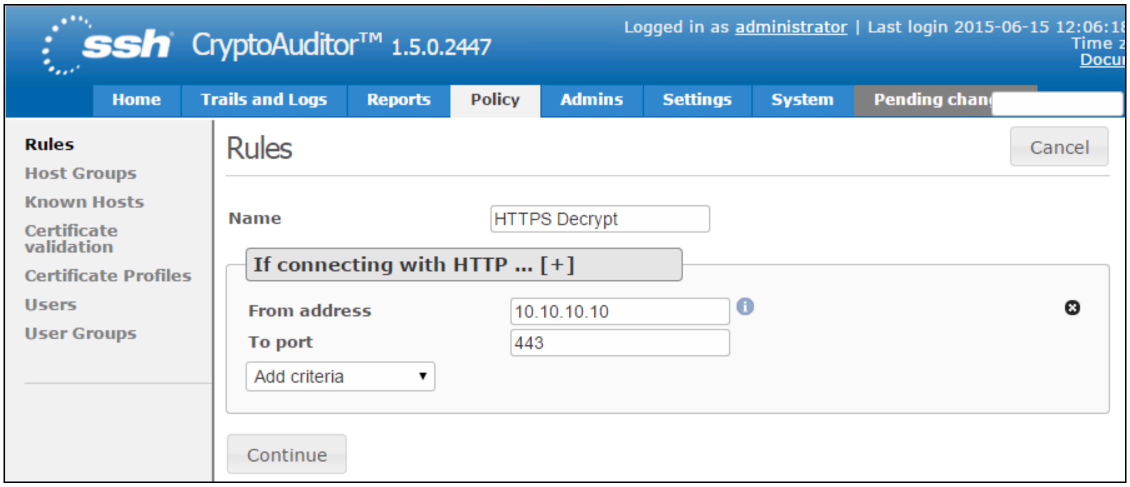
1. In the CryptoAuditor Admin UI, navigate to **Policy… Rules…+Add Rule**:



Rules

HTTP SSL Decrypt
from IP 10.10.10.10 to port 443
to all users
all [full]

2. Give the rule a name and select **HTTP** in the **If connecting with…** dialog. Then choose the address or group of address you wish to audit, and enter 443 as the **To port**.



Rules

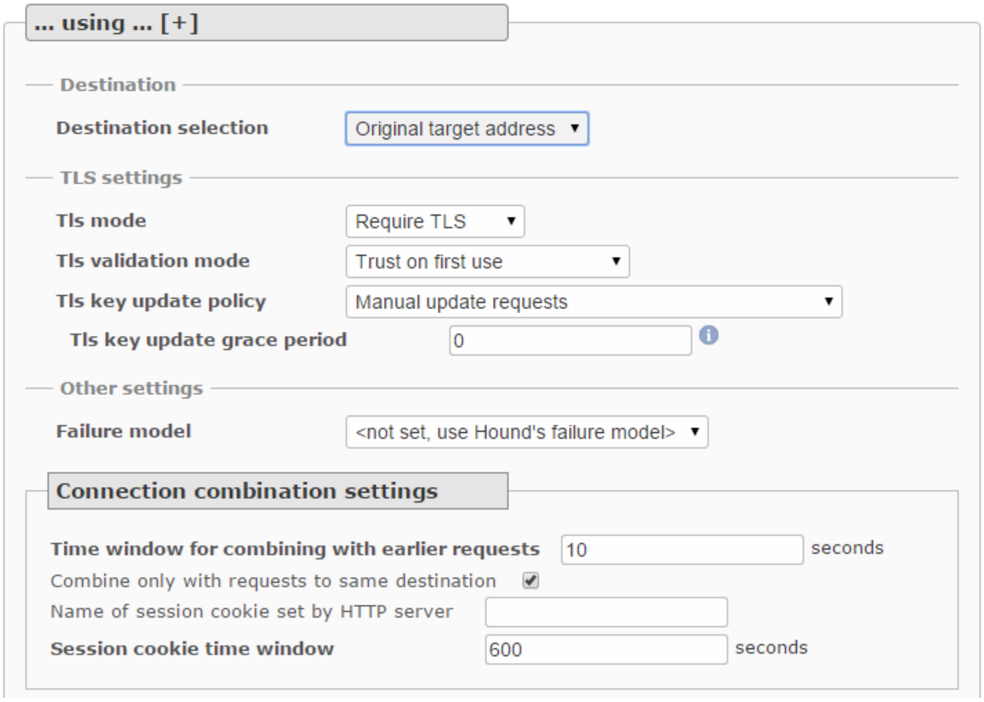
Name: HTTPS Decrypt

If connecting with HTTP ... [+]

From address: 10.10.10.10

To port: 443

3. Click **Continue** to see the **…using…** dialog. Under the TLS settings, you **must** choose **Require TLS** in order to audit HTTPS sessions.



… using … [+]

Destination

Destination selection: Original target address ▾

TLS settings

Tls mode: Require TLS ▾

Tls validation mode: Trust on first use ▾

Tls key update policy: Manual update requests ▾

Tls key update grace period: 0

Other settings

Failure model: <not set, use Hound's failure model> ▾

Connection combination settings

Time window for combining with earlier requests: 10 seconds

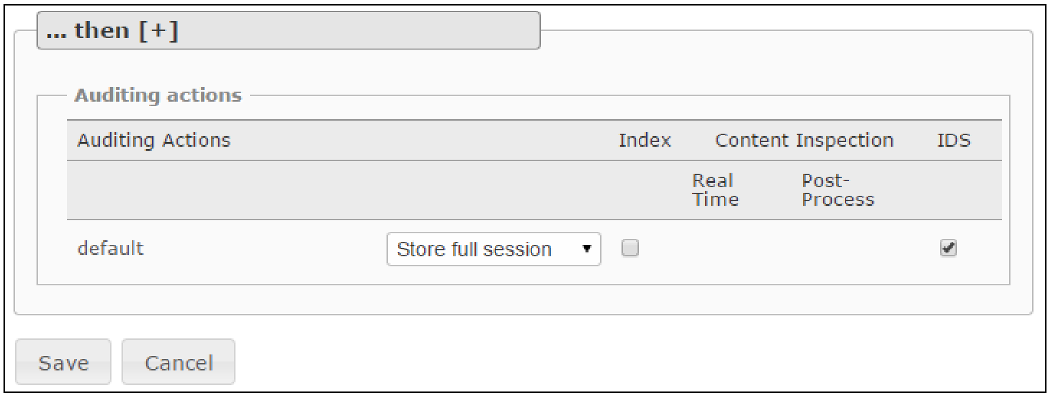
Combine only with requests to same destination ☑

Name of session cookie set by HTTP server:

Session cookie time window: 600 seconds

Other paramters may be configured as outlined in the CryptoAuditor online help.

4. Under the **… then [+]** heading, choose **Store full session** as the auditing action. Select the **IDS** checkbox to instruct CryptoAuditor to have CryptoAuditor direct the decrypted traffic to the IDS interface. This interface should be on the same network as your RSA Security Analytics Packet Decoder.



… then [+]

Auditing actions

| Auditing Actions | | Index | Content Inspection | | IDS |
|---|---|---|---|---|---|
| | | | Real Time | Post-Process | |
| default | Store full session ▾ | ☐ | | | ☑ |

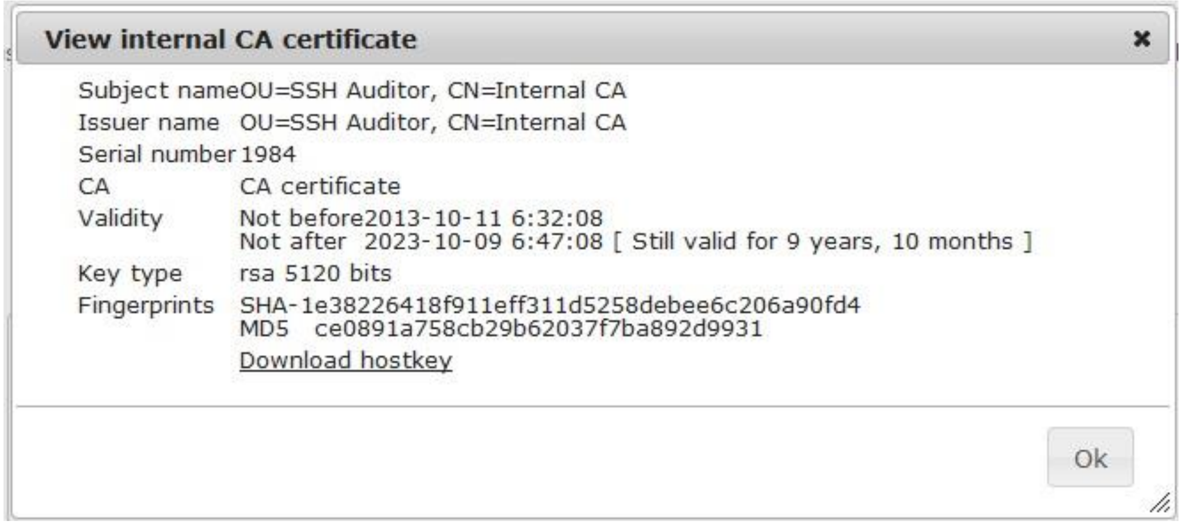
5. Click **Sav**e to save the connection rule. Now that the rule has been created, you can configure the IDS interface for the CryptoAuditor Hound to ensure proper commmunication with the RSA Security Analytics Packet Decoder.

   Now that the Connection Rule has been created, and the IDS interface has been configured, it is necessary to distribute the **Internal CA Certificate** to client systems.

### Distributing the CryptoAuditor Internal CA Certificate

You can view and download the CryptoAuditor's internal CA certificate, but you cannot change or remove it. You may want to download the certificate in case that you need to install the certificate to client systems using HTTPS.  In audited HTTPS connections, CryptoAuditor uses the internal CA certificate for generating temporary host certificates for the connection between the HTTPS client and the Hound. Because CryptoAuditor's internal CA is not a trusted authority in the HTTPS clients (for example a web browser), you will need to install it to the client hosts and applications as a trusted authority.

To view the certificate information, navigate to **Policy… Certificate Validation**, and click **Internal CA Certificate** at the end of the page.



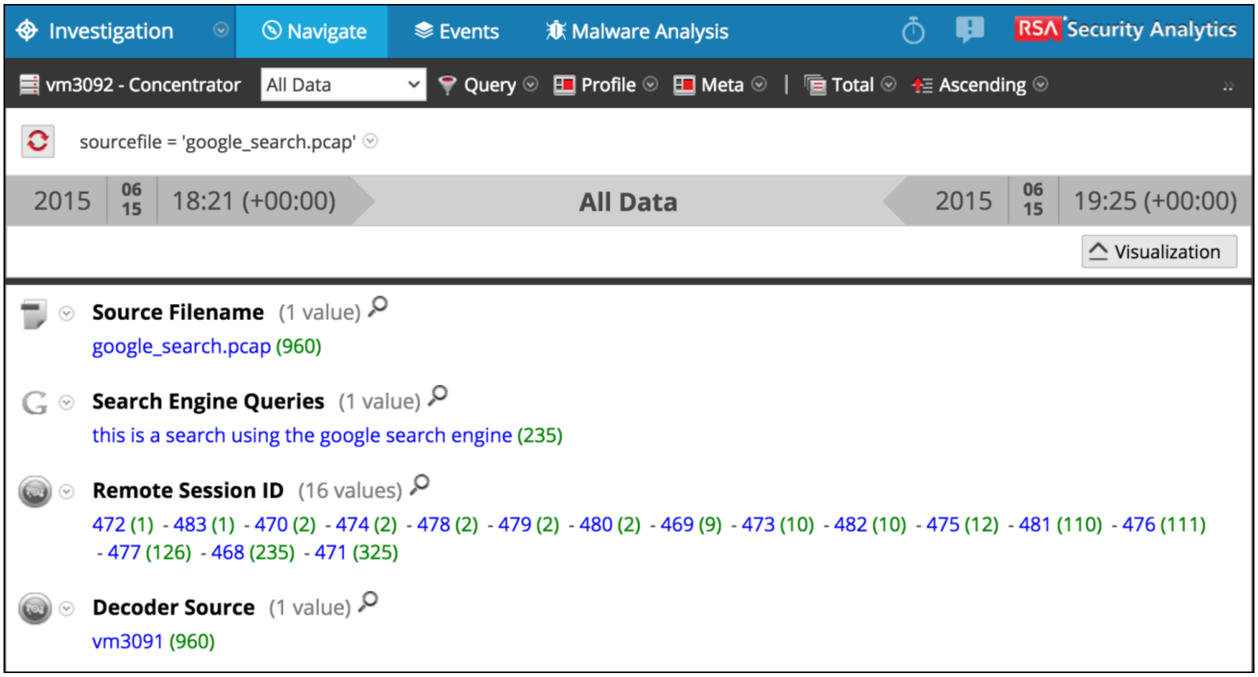
| | |
|---|---|
| Subject name | OU=SSH Auditor, CN=Internal CA |
| Issuer name | OU=SSH Auditor, CN=Internal CA |
| Serial number | 1984 |
| CA | CA certificate |
| Validity | Not before2013-10-11 6:32:08<br>Not after  2023-10-09 6:47:08 [ Still valid for 9 years, 10 months ] |
| Key type | rsa 5120 bits |
| Fingerprints | SHA-1e38226418f911eff311d5258debee6c206a90fd4<br>MD5   ce0891a758cb29b62037f7ba892d9931<br>Download hostkey |

To download the certificate, in the **View internal CA certificate** window, click **Download hostkey.**

## *RSA Security Analytics Configuration*

There is no additional configuration required in RSA Security Analytics other than to have to appropriate parsers installed in order to analyze the type of SSL traffic you are decrypting.

Once the Security Analytics Packet Decoder has captured traffic and sent it to the Concentrator, you should see the related metadata when performing an investigation -- for example, a decrypted Google Search using the **Search_Engines** parser:



sourcefile = 'google_search.pcap'

2015 06 15 18:21 (+00:00) All Data 2015 06 15 19:25 (+00:00)

**Source Filename** (1 value)
google_search.pcap (960)

**Search Engine Queries** (1 value)
this is a search using the google search engine (235)

**Remote Session ID** (16 values)
472 (1) - 483 (1) - 470 (2) - 474 (2) - 478 (2) - 479 (2) - 480 (2) - 469 (9) - 473 (10) - 482 (10) - 475 (12) - 481 (110) - 476 (111) - 477 (126) - 468 (235) - 471 (325)

**Decoder Source** (1 value)
vm3091 (960)

# Certification Checklist for RSA Security Analytics

Date Tested: June 15th, 2015

| Certification Environment | | |
|---|---|---|
| **Product Name** | **Version Information** | **Operating System** |
| **RSA Security Analytics** | 10.4.1 | Virtual Appliance |
| **SSH CryptoAuditor** | 1.5.0.2447 | Virtual Appliance |
| | | |

| Security Analytics Test Case | Result |
|---|---|
| **Outbound SSL Decryption** | |
| **HTTPS** | |
| Google Search | ✓ |
| Bing Search | ✓ |
| Facebook | ✓ |
| YouTube | ✓ |
| Twitter | ✓ |
| LinkedIn | ✓ |
| Reddit | ✓ |
| **WEBMAIL** | |
| GMail | ✓ |
| Yahoo | ✓ |
| Live | ✓ |
| AOL | ✓ |
| **Inbound SSL Decryption** | |
| **HTTPS** | |
| Web Server | N/A |

JEC

✓ = Pass ✗ = Fail N/A = Non-Available Function