National Information Assurance Partnership

Common Criteria Evaluation and Validation Scheme
Validation Report

Information Assurance Specialists, Inc.

IAS Router

Report Number: CCEVS-VR-VID10625-2015
Dated: December 21, 2015
Version: 1.0

National Institute of Standards and Technology
Information Technology Laboratory
100 Bureau Drive
Gaithersburg, MD 20899

National Security Agency
Information Assurance Directorate
9800 Savage Road STE 6940
Fort George G. Meade, MD 20755-6940

Acknowledgements

Validation Panel

Patrick Mallett

The Mitre Corporation, McLean, VA

Jerome Myers

The Aerospace Corporation, Columbia, MD

Common Criteria Testing Laboratory

Brad Mitchell, Ryan Day

InfoGard Laboratories, Inc.
San Luis Obispo, CA

Table of Contents

1 Executive Summary

This report documents the NIAP validators' assessment of the CCEVS evaluation of the IAS Router by Information Assurance Specialists, Inc.

This report is intended to assist the end-user of this product with determining the suitability of this IT product in their environment. End-users should review both the Security Target (ST), which is where specific security claims are made, in conjunction with this Validation Report (VR), which describes how those security claims were evaluated.

The Target of Evaluation (TOE) is the IAS Router; a VPN Gateway Network Device. The IAS Routers are a family of ultra-portable routers that offer advanced routing capabilities and diverse WAN technology options which enables the users to leverage a wide array of WAN technologies (e.g. Ethernet, Wi-Fi, Cellular) to establish a VPN connection between the IAS Router and a secure LAN.

This table identifies components that must be present in the Operational Environment to support the operation of the TOE:

Component	Description
Syslog Server	An RFC 5424 compliant syslog server supporting IPsec connections to support storage and review of audit logs
NTP Server (Optional)	The TOE supports syncing time with an NTP Server: • NTPv4 (RFC 5905)
VPN Peer	A VPN Peer supporting: • IPsec/IKEv1 (RFCs 2407, 2408, 2409, 4109) & IKEv2 (RFC 5996) o Main Mode o Authentication with X.509 using: ▪ ECDSA (P-256 or P-384) ▪ Pre-Shared Key o Symmetric ciphers (at least one of): ▪ AES-CBC-128 ▪ AES-CBC-256 o Integrity Algorithms (at least one of): ▪ HMAC-SHA-256 ▪ HMAC-SHA-384 ▪ HMAC-SHA-512 o Key Agreement (at least one of): ▪ Diffie-Hellman Group 14 (2048 modp) ▪ Diffie-Hellman Group 19 (P-256) ▪ Diffie-Hellman Group 20 (P-384) ▪ Diffie-Hellman Group 21 (P-521)

	• IPsec/ESP (RFCs 4301, 4303, 4106, 3602): o Tunnel Mode o Symmetric ciphers (at least one of): ▪ AES-GCM-128 ▪ AES-GCM-256 ▪ AES-CBC-128 ▪ AES-CBC-256 o Integrity (only with AES-CBC, at least one of): ▪ HMAC-SHA-256 ▪ HMAC-SHA-384 ▪ HMAC-SHA-512
Serial Connection	RS-232 serial connection for local console administration
Web browser (optional)	For local or remote administration, a web browser of the following characteristics can be utilized: The TOE is known to be compatible with IE 10+, Chrome 29+, Firefox 22+, and Safari 6+. The TOE requires a Web Browser (Remote Console) supporting: • Protocol versions (at least one of): o HTTPs/TLSv1.0 (RFCs 2818 & 2246) o HTTPs/TLSv1.1 (RFC 2818 & 3246) o HTTPs/TLSv1.2 (RFCs 2818 & 5246) • Ciphersuites (at least one of): o TLS_RSA_WITH_AES_128_CBC_SHA o TLS_RSA_WITH_AES_256_CBC_SHA o TLS_DHE_RSA_WITH_AES_128_CBC_SHA o TLS_DHE_RSA_WITH_AES_256_CBC_SHA o TLS_RSA_WITH_AES_128_CBC_SHA256 o TLS_RSA_WITH_AES_256_CBC_ SHA256 o TLS_DHE_RSA_WITH_AES_128_CBC_ SHA256 o TLS_DHE_RSA_WITH_AES_256_CBC_ SHA256 o TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA256 o TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA384 o TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256 TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384

Table 1: Operational Environment Components

2 Identification of the TOE

Table 2 provides information needed to completely identify the product, including:

- The Target of Evaluation (TOE), the fully qualified identifier of the product as evaluated;
- The Security Target (ST), describing the security features, claims, and assurances of the product;
- The conformance result of the evaluation;
- The organizations and individuals participating in the evaluation.

Evaluation Scheme	United States Common Criteria Evaluation Validation Scheme
Evaluated Target of Evaluation	IAS STEW Rev. 1.0, with IASRouter-2015-11-24_50e8756_Release-x86-fips_cc.firmware
Protection Profile	• Protection Profile for Network Devices, Version 1.1, June 8, 2012 • Security Requirements for Network Devices Errata #3, November 3, 2014 • Network Device Protection Profile Extended Package VPN Gateway, Version 1.1, April 12, 2013
Security Target	IAS Router Security Target, Version 1.0, December 21, 2015
Dates of Evaluation	February 23 – December 7, 2015
Conformance Result	Pass
Common Criteria Version	Version 3.1 Revision 3, July 2009
Common Evaluation Methodology (CEM) Version	Version 3.1, Revision 3, July 2009
Evaluation Technical Report (ETR)	15-3348-R-0039, Version 1.1, December 21, 2015
Sponsor/Developer	Information Assurance Specialists, Inc.
Common Criteria Testing Lab (CCTL)	InfoGard Laboratories, Inc.
CCTL Evaluators	Brad Mitchell, Ryan Day
CCEVS Validators	Patrick Mallett, Jerome Myers

Table 2: Product Identification

3 Interpretations

The Evaluation Team performed an analysis of the international interpretations of the CC and the CEM and determined that none of the International interpretations issued by the Common Criteria Interpretations Management Board (CCIMB) were applicable to this evaluation.

The TOE is also compliant with all international interpretations with effective dates on or before June 2, 2015.

4 Security Policy

This section contains the product features and denotes which are within the logical boundaries of the TOE. The following Security Functions are supported by the TOE:

- Audit
- Cryptography
- User Data Protection
- Identification and Authentication
- Security Management
- Packet Filtering
- Protection of the TSF
- TOE Access
- Trusted Path/Channels

4.1 Audit

The TSF generates and formats audit records according to the Syslog Protocol (RFC 5424). These records include time stamp, hostname or IP address, process name, user identifier, action, and target. For administrative actions, the TSF logs the administrator's username as the user identifier. For network actions, the user identifier is the IP address or interface that triggered the event.

Audit records sent to the remote server are protected by an IPsec tunnel. The TOE prevents modification to the local audit log. If the IPsec tunnel has not been established, the TSF does not transmit audit logs. The TOE maintains a local audit log in addition to sending the audit records to a remote Syslog server. The TSF maintains 1 GB of local audit log files in a RAM disk (volatile), so all local audit logs are lost if power is lost.

The TSF performs log rotation if the current log file is greater than 1 MB. Log rotation involves compressing the current log file and archiving it. The TSF then adds all future audit events to a new log file. The TSF deletes the oldest log archive when there are more than four archives or if local storage is exhausted.

4.2 Cryptographic Operations

4.2.1 Cryptographic Certifications

The TOE implements CAVP validated cryptographic algorithms for random bit generation, encryption/decryption, authentication, and integrity protection/verification. These algorithms are used to provide security for the TLS and IPsec (IKEv1, IKEv2, and ESP) protocols.

The TSF implements the following CAVP validated algorithms:

- AES (Cert. #3430)
- RSA (Cert. #1756)
- DSA (Cert. #948)
- ECDSA (Cert. #663)
- SHA (Cert. #2830)
- HMAC (Cert. #2182)
- DRBG (Cert. #782)

These algorithms are implemented in the IAS-Router-FIPS crypto library version 7a55571 – 2015-05-07. This library is part of IASRouter-2015-11-24_50e8756_Release-x86-fips_cc.firmware.

4.2.2 Zeroization

The TOE zeroizes all plaintext secret and private cryptographic keys and CSPs once they are no longer required.

The TSF maintains the following persistent secret and private keys in the file system (Flash/SSD):

- IPsec ECDSA and/or RSA private key
- IPsec pre-shared key
- TLS ECDSA and/or RSA private key

The TSF zeroizes persistent CSPs whenever a file containing CSPs is modified or deleted by overwriting the specified file three times with a pseudo random pattern.

The TSF maintains the following secret and private keys in volatile memory (RAM):

- IKE DH or ECDH Private Key
- IKE Session Keys
- ESP Session Keys
- TLS DH or ECDH Private Key
- TLS pre-master secret & TLS master secret
- TLS Session Keys

The TSF zeroizes volatile secret and private keys when power is removed[1].

When the user invokes a “Reset to factory defaults,” the TSF performs a zeroization of all persistent CSPs, followed by a system reboot to zeroize all volatile CSPs.

4.2.3 Random Bit Generation

The TSF implements an SP 800-90A CTR_DRBG (AES 256) for generating random bits. The correct operation of the DRBG has been validated by the Cryptographic Algorithm Validation Program (DRBG Cert #TBD782).

[1] This method of zeroization meets the NSA CSS Storage Device Declassification Manual for the zeroization of DRAM and SRAM.

4.2.4 IPsec

The TSF implements IPsec as specified in RFCs 4301, 4303, 4106, and 3602. The TSF supports IPsec connections operating in tunnel mode.

The TSF can be configured to use the AES-GCM-128, AES-GCM-256, AES-CBC-128 with an HMAC, and/or AES-CBC-256 with an HMAC as the encryption and message authentication algorithms for IPsec ESP. When AES-CBC is negotiated as the symmetric cipher, the TOE supports HMAC-SHA-256, HMAC-SHA-384, and HMAC-SHA-512.

The TSF implements IKEv1 as specified in RFCs 2407, 2408, 2409, and 4109 and IKEv2 as specified in RFC 6379. The TSF uses AES-CBC-128 or AES-CBC-256 to encrypt the IKE payloads.

The TSF can be configured to use the following SHA-based HMAC algorithms in IKEv1 or IKEv2:

- SHA-256
- SHA-384
- SHA-512

The TSF supports DH groups 14, 19, 20 and 21 for use in IKE. The TSF supports ECDSA x.509 certificates and Pre-Shared Keys to perform IKE peer authentication. The ECDSA certificates must use “NIST curves” P-256 or P-384.

4.2.5 TLS

The TSF implements the server side of TLSv1.0, TLSv1.1, and TLSv1.2 according to RFCs 2246, 4346, and 5246 respectively. The TSF also implements the extension specified in RFCs 3286, 4301, and 5289.

The TSF supports the following TLS cipher suites:

- TLS_RSA_WITH_AES_128_CBC_SHA
- TLS_RSA_WITH_AES_256_CBC_SHA
- TLS_DHE_RSA_WITH_AES_128_CBC_SHA
- TLS_DHE_RSA_WITH_AES_256_CBC_SHA
- TLS_RSA_WITH_AES_128_CBC_SHA256
- TLS_RSA_WITH_AES_256_CBC_ SHA256
- TLS_DHE_RSA_WITH_AES_128_CBC_ SHA256
- TLS_DHE_RSA_WITH_AES_256_CBC_ SHA256
- TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA256
- TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA384
- TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256
- TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384

The TSF only supports the AES GCM cipher suites when TLSv1.2 is negotiated.

4.2.6 HTTPS

The TSF implements the server side of the HTTPs protocol according to RFC 2818 by using a TLS connection in place of a TCP connection. The TSF uses HTML over HTTPs to present the administrative users with a secure management interface. The TSF uses TLS to provide a secure connection between the TSF and the administrator; however, HTTP is used to maintain the administrator's session. The management interface performs administrator authentication.

4.3 User Data Protection

The TSF ensures that data will not be reused when processing network packets by overwriting previous buffer contents upon allocation of a new buffer. The TSF accomplishes this by allocating the exact buffer size for each write/addition to each buffer. This ensures that the previous contents are immediately overwritten. The TSF continues to enlarge the buffer by the exact size of each additional write to progressively "grows" the buffer as necessary. The TOE zeroizes packet buffers when each buffer is allocated.

4.4 Identification and Authentication

The TOE authenticates administrative users using a username/password combination. The TSF does not allow access to any administrative functions prior to successful authentication. The TOE has the capability to lock a remote user's account if that user exceeds the configured number of failed authentication attempts.

The TSF provides local console and the HTTPs web GUI to administer the TSF.

When a user connects to the console interface, the TSF prompts the user for a username and password. The TSF does not echo any characters back to the local console while the user is entering their password.

When a user connects to the HTTPS interface, the TSF prompts the user for a username and password. The TSF presents the user's browser with an HTML Password field to indicate that the characters should not be echoed back; however, displaying or hiding the password is outside of the control of the TSF.

The TSF requires passwords to be 15 characters or greater. The TSF supports passwords containing lowercase, uppercase, and numeric ASCII characters, including the following special characters: !@#$%^&*()

The TSF supports text-based pre-shared keys for authentication of IKE authentication. The TSF allows the entry of text-based pre-shared keys that are at least 15 characters in length. The TSF uses the IKE negotiated HMAC function to condition pre-shared keys.

The TSF also supports the entry of bit-based pre-shared keys. The minimum length of a decoded bit-based pre-shared key is 112-bits.

The TSF maintains a counter of failed remote authentication attempts in volatile memory. The TSF increments this counter each time an incorrect username/password combination is submitted over HTTPs. When the counter reaches a configured value (default 5), the TSF blocks all remote authentication attempts for a configurable period of time (default 10 minutes).

The administrative interfaces do not implement commands to allow unauthorized users to the certificate store. The TSF also zeroizes certificates according to the persistent zeroization. The TSF allows the administrator to load certificates with a specific command over HTTPs.

The TSF verifies the validity of a certificate when an administrator loads a certificate into the TSF and when IKE receives a peer certificate. If the administrator attempts to load a certificate with a Subject Type=CA, the TSF does not validate the certificate path.

4.5 Security Management

The TSF implements two security management interfaces, a limited local console and a HTML based GUI. Regardless of interface, the TSF does not allow any administrative actions to be performed prior to authentication of the administrative user.

The GUI provides the administrator with additional administrative functions; however, the functions are limited to those explicitly presented by the GUI to prevent the user from running arbitrary commands.

4.6 Packet Filtering

The TOE filters packets received on the physical interfaces and virtual interfaces (IPsec tunnels). The TOE reads each packet's header and can be configured to allow or deny the packet based on IP source address, IP destination address, Transport Layer Protocol (if specified in the IP header), TCP or UDP source port, and/or TCP or UDP destination port.

While the TSF is powering up, the TSF starts security functions in the following order: audit, POST, packet filtering, networking, HTTPs administration, VPN. This ensures that the TSF has passed its self-tests and enabled packet filtering before any network interfaces are enabled. If the POST fails or the packet filtering rules fail to start, the TSF automatically reboots without initializing networking. This ensures that the TSF is operating properly and that the packet filtering rules have been initialized before the TSF processes any network data.

The TSF has the following zones:

- WAN: 1xRJ45 or internal connector to the Cisco ESR
- VPN: Internal Virtual Interface
- LAN: 5xRJ45

The TSF ensures that the rules are not bypassed in the event of a component failure by utilizing the three zones. The FORWARD rules specify a zone instead of a hardware interface, so failure of the VPN results in iptables being unable to send data to the VPN zone (from the WAN or LAN zones). iptables is the only process that passes traffic between the difference zones, so a failure of iptables results in the inability to pass any traffic.

The TSF performs stateful packet inspection to determine if a packet is part of an established TCP stream. The TSF performs this inspection by checking the source address, destination address, source port, destination port, sequence number against, and flags against established streams. If the TSF determines that the packet is part of an established stream, it forwards or accepts the packet without applying all of the rules. If a packet is not part of an established TCP session, the TSF applies the rules sequentially. If a packet matches a rule, the action configured in the rule is applied. If the packet does not match the rule or the action was only LOG, the TSF passes the packet to the next rule in the chain. The TSF implements a hard-coded rule at the end of each chain that cannot be modified:

- REJECT and LOG any packets that do not match any user created rule.

The IPv4, TCP, and UDP protocols implemented by the TSF have been verified to conform with RFCs 791, 793, and 768 respectively. Conformance with these RFCs has been tested by interoperability testing with the Windows 7, MacOS X 10.8, CentOS 6, and Cisco IOS 15 network stacks.

4.7 Protection of the TSF

The TOE protects itself through a number of features. The administrative interfaces do not allow the administrator to execute arbitrary binaries or provide commands for the administrator to display secret and private keys. The TOE ensures timestamps and timeouts are accurate by maintaining a real-time clock for measuring time as well as polling an NTP server to mitigate drift.

The TOE implements self-tests to verify its correct operation prior to enabling networking. While the TSF is powering up and performing its power-up self-tests, it does not pass any network traffic. If the power-on self-tests fail or a fatal conditional self-test fails, the TOE enters an error state, disables network services, and disables all cryptographic operations.

The TOE automatically verifies the authenticity and integrity of updates by requiring the updates to be digitally signed. The TOE verifies that every update is digitally signed prior to installing the update.

4.8 TOE Access

The administrator can access the TSF via the local console (serial) or remotely via HTTPs. The TSF displays a configurable advisory and consent message when an administrator accesses the local console or HTTPs interface. The administrator can terminate a console or HTTPs session by logging out. When an administrator logs out of the local console, the TSF sets the state to unauthenticated and presents a login prompt. When an administrator logs out of the HTTPs session, the TSF sets a flag in the HTTPs session to unauthenticated. All HTTPs sessions are cleared when the TSF is shutdown or restarted. The TSF terminates local console and HTTPs sessions after a configurable period of inactivity. The TSF immediately terminates local sessions if the period of inactivity expires. The TSF computes the time from the last activity to the current request upon receipt of each HTTPs request. If the time difference exceeds the inactivity timer, the TSF does not process the request and terminates the session. The TOE can be configured to deny establishment of a VPN client session based on the time, day, and/or remote client's IP address.

4.9 Trusted Path/Channels

The TOE uses IPsec to provide a trusted communication channel between itself and VPN peers. The trusted channels utilize X.509 certificates or pre-shared keys to perform mutual authentication. The TOE initiates the IPsec trusted channel with a remote peer to protect user data and protect communication with the syslog server.

The TOE uses TLS/HTTPs to provide a trusted path between itself and remote administrative users. The TOE does not implement any additional methods of remote administration. The administrator can configure the remote administration to be tunneled through IPsec in addition to using TLS/HTTPs.

5 TOE Security Environment

5.1 Secure Usage Assumptions

The following assumptions are made about the usage of the TOE:

A.NO_GENERAL_PURPOSE	It is assumed that there are no general-purpose computing capabilities (e.g., compilers or user applications) available on the TOE, other than those services necessary for the operation, administration and support of the TOE.
A.PHYSICAL	Physical security, commensurate with the value of the TOE and the data it contains, is assumed to be provided by the environment.
A.TRUSTED_ADMIN	TOE Administrators are trusted to follow and apply all admin guidance in a trusted manner.
A.CONNECTIONS	It is assumed that the TOE is connected to distinct networks in a manner that ensures that the TOE security policies will be enforced on all applicable network traffic flowing among the attached networks.

5.2 Threats Countered by the TOE

The TOE is designed to counter the following threats:

T.ADMIN_ERROR	An authorized administrator may incorrectly install or configure the TOE incorrectly, resulting in ineffective security mechanisms.
T.TSF_FAILURE	Security mechanisms of the TOE may fail, leading to a compromise of the TSF.
T.UNDETECTED_ACTIONS	Malicious remote users or external IT entities may take actions that adversely affect the security of the TOE. These actions may remain undetected and thus their effects cannot be effectively mitigated.
T.UNAUTHORIZED_ACCESS	A user may gain unauthorized access to the TOE data and TOE executable code. A malicious user, process, or external IT entity may masquerade as an authorized entity in order to gain unauthorized access to data or TOE resources. A malicious user, process, or external IT entity may misrepresent itself as the TOE to obtain identification and authentication data.
T.UNAUTHORIZED_UPDATE	A malicious party attempts to supply the end user with an update to the product that may compromise the security features of the TOE.
T.USER_DATA_REUSE	User data may be inadvertently sent to a destination not intended by the original sender.
T.NETWORK_DISCLOSURE	Sensitive information on a protected network might be disclosed resulting from ingress- or egress-based actions.
T.NETWORK_ACCESS	Unauthorized access may be achieved to services on a protected network from outside that network, or alternately services outside a protected network from inside the protected network
T.NETWORK_MISUSE	Access to services made available by a protected network might be used counter to Operational Environment policies.
T.REPLAY_ATTACK	If malicious or external IT entities are able to gain access to the network, they may have the ability to capture information traversing throughout the network and send them on to the intended receiver.

T.DATA_INTEGRITY	A malicious party attempts to change the data being sent – resulting in loss of integrity.

5.3 Organizational Security Policies

The TOE enforces the following OSPs:

P.ACCESS_BANNER	The TOE shall display an initial banner describing restrictions of use, legal agreements, or any other appropriate information to which users consent by accessing the TOE.

5.4 Security Objectives

The following are security objectives of the TOE:

O.PROTECTED_COMMUNICATIONS	The TOE will provide protected communication channels for administrators, other parts of a distributed TOE, and authorized IT entities.
O.VERIFIABLE_UPDATES	The TOE will provide the capability to help ensure that any updates to the TOE can be verified by the administrator to be unaltered and (optionally) from a trusted source.
O.SYSTEM_MONITORING	The TOE will provide the capability to generate audit data and send those data to an external IT entity.
O.DISPLAY_BANNER	The TOE will display an advisory warning regarding use of the TOE.
O.TOE_ADMINISTRATION	The TOE will provide mechanisms to ensure that only administrators are able to log in and configure the TOE, and provide protections for logged-in administrators.
O.PROTECTED_COMMUNICATIONS	The TOE will provide protected communication channels for administrators, other parts of a distributed TOE, and authorized IT entities.
O.VERIFIABLE_UPDATES	The TOE will provide the capability to help ensure that any updates to the TOE can be verified by the administrator to be unaltered and (optionally) from a trusted source.
O.SYSTEM_MONITORING	The TOE will provide the capability to generate audit data and send those data to an external IT entity.
O.DISPLAY_BANNER	The TOE will display an advisory warning regarding use of the TOE.
O.TOE_ADMINISTRATION	The TOE will provide mechanisms to ensure that only administrators are able to log in and configure the TOE, and provide protections for logged-in administrators.
O.RESIDUAL_INFORMATION_CLEARING	The TOE will ensure that any data contained in a protected resource is not available when the resource is reallocated.
O.SESSION_LOCK	The TOE shall provide mechanisms that mitigate the risk of unattended sessions being hijacked.

O.TSF_SELF_TEST	The TOE will provide the capability to test some subset of its security functionality to ensure it is operating properly.
O.ADDRESS_FILTERING	The TOE will provide the means to filter and log network packets based on source and destination addresses.
O.AUTHENTICATION	The TOE will provide a means to authenticate the user to ensure they are communicating with an authorized external IT entity.
O.CRYPTOGRAPHIC_FUNCTIONS	The TOE will provide means to encrypt and decrypt data as a means to maintain confidentiality and allow for detection and modification of TSF data that is transmitted outside of the TOE.
O.FAIL_SECURE	Upon a self-test failure, the TOE will shutdown to ensure data cannot be passed while not adhering to the security policies configured by the administrator.
O.PORT_FILTERING	The TOE will provide the means to filter and log network packets based on source and destination transport layer ports.

5.5 Security Objectives for the Operational Environment

The following are security objectives of the Operational Environment:

OE.NO_GENERAL_PURPOSE	There are no general-purpose computing capabilities (e.g., compilers or user applications) available to the TOE, other than those services necessary for the operation, administration and support of the TOE.
OE.PHYSICAL	Physical security, commensurate with the value of the TOE and the data it contains, is provided by the environment.
OE.TRUSTED_ADMIN	TOE Administrators are trusted to follow and apply all administrator guidance in a trusted manner.
OE.CONNECTIONS	TOE administrators will ensure that the TOE is installed in a manner that will allow the TOE to effectively enforce its policies on network traffic flowing among attached networks.

6 Architectural Information

The TOE is classified as a VPN Gateway Network Device for Common Criteria purposes. The TOE is made up of hardware and software components.

6.1 Architecture Overview

The Target of Evaluation (TOE) is a VPN Gateway Network Device (router) and consists of the following hardware:

- IAS STEW Rev. 1.0
- IAS KG-RU Rev. 1.0
- IAS Router Micro Rev. 1.0

running:

- IASRouter-2015-11-24_50e8756_Release-x86-fips_cc. firmware

6.1.1 TOE Hardware

The IAS STEW and IAS KG-RU 2015 include a Cisco ESR 5915 within the physical enclosure; however, the IAS Router is an independently configured, evaluated, and tested component that does not have any security dependencies on the Cisco ESR.

The IAS STEW, IAS KG-RU, and IAS MICRO devices appear superficially as unique device form factors, yet all three devices use common hardware under their respective enclosures. The three devices leverage a technology known as Computer on Module Express, or COM E for short. The concept of COM E is that a developer can create different application specific “carrier cards” to meet their I/O interface requirements by leveraging an off-the-shelf COM E module to perform the computing tasks. The IAS STEW and IAS Router MICRO use the same COM E carrier card, the difference being that the IAS STEW has an additional “carrier card” for an additional component (Cisco ESR5915) used in the IAS STEW enclosure (not part of the NIAP CC PP validation effort). The KG-RU uses a COM E carrier card that was designed to retain both the COM E module AND the Cisco ESR5915 module on a single carrier board versus the two separate carriers used in the IAS STEW.

Components used across the IAS STEW, IAS KG-RU, and IAS MICRO devices are nearly identical, yet PCB design (and subsequent PCB traces) is unique to the COM E carrier card used within the IAS Router MICRO and IAS STEW versus the COM E carrier card used within the KG-RU. All three devices use the same Adlink cExpress BT-E3845 COM Express module that uses the Intel E3845 Baytrail Atom processor. The IAS Router MICRO uses this same COM E module and the same COM E carrier card PCB from within the IAS STEW (but the MICRO omits the use of the additional Cisco carrier board found in the IAS STEW). The Intel E3845 Baytrail processor is a System-On-a-Chip, and so there is no additional processor south bridge components in comparison to other Intel based X86 processors and computing systems. Each of the three devices uses the Intel 82000 series GbE controller for the numerous additional Ethernet ports (over and above what the Intel E3845 SOC offers).

All three devices use mSATA SSD drives for their firmware storage, and standard 1333/1066 Mhz DDR3L RAM in SODIMM sockets. Functionally, the IAS STEW, KG-RU and IAS Router MCIRO use the exact same computing and network interface components.

6.1.2 TOE Software

Just as the IAS STEW, KG-RU and IAS Router MICRO are the same insofar as hardware, they also run the same IAS Router Firmware. The IAS Router Firmware was designed in such a way that it is common across the three hardware devices.

The TOE operates on the IASRouter-2015-11-24_50e8756_Release-x86-fips_cc. firmware.

7 Documentation

This section details the documentation that is either delivered to the customer, and/or was used as evidence for the evaluation of the IAS Router.

7.1 Guidance Documentation

Document	Revision	Date
IAS Router Common Criteria Operator Guidance	1.0.8	December 21, 2015
Hardware Manual for the IAS KG-RU Communications Solution	0.2	June 27, 2015
Hardware Manual for the IAS Router Micro	0.2	June 27, 2015
Hardware Manual for the IAS Small Tactical Executive WAN (STEW) Communications Solution	0.2	June 27, 2015

7.2 Test Documentation

Document	Revision	Date
15-3348-R-0013 V1.4 IAS VPN Test Plan	1.4	December 21, 2015

7.3 Security Target

Document	Revision	Date
IAS Router Security Target	1.0	December 21, 2015
IAS Router Entropy Rationale and Randomizer Design Details	1.1	N/A

8 IT Product Testing

This section describes the testing efforts of the Developer and the Evaluation Team.

8.1 Evaluation Team Independent Testing

The Evaluation team at the CCTL (InfoGard Laboratories, Inc.) generated the testing plan and designed the testing activities specified in the Protection Profile for Network Devices, Version 1.1, June 8, 2012, the Security Requirements for Network Devices Errata #3, November 3, 2015, and the Network Device Protection Profile Extended Package VPN Gateway, Version 1.1, April 12, 2013, generating automated and manual tests to execute the designed test plan. The testing activities were conducted as specified in the Protection Profile for Network Devices, Version 1.1, June 8, 2012, the Security Requirements for Network Devices Errata #3, November 3, 2015, and the Network Device Protection Profile Extended Package VPN Gateway, Version 1.1, April 12, 2013.

8.2 Vulnerability Analysis

The evaluator generated network packets that tested all of the values forType, Code, and Transport Layer Protocol that are undefined by the RFC for each of the protocols, ICMPv4, and IPv4.

For example, ICMPv4 has an eight-byte field for Type and an eight-byte field for the Code. Only 21 Types are defined in the RFC, but there are 256 possible values. Each Type has a Code associated with it, the number of RFC defined Codes varies based on the Type. The evaluator constructed packets that tested each possible value not defined in the RFC (the defined values are already tested in FPF_RUL_EXT.1.10) of Type and Code (including all possible combinations) and targeted each distinct interface type to determine that the TOE handles these packets appropriately. Since none of these packets matched a rule, or belonged to an allowed session the packets were appropriately dropped. Since there are no requirements that the firewall audit a packet being dropped under these circumstances, the evaluator ensured that the firewall did not allow these packets to flow through the TOE.

In addition to the undefined attribute testing required above, the evaluator performed intelligent fuzz testing of the remaining fields in the required protocol headers (excluding FTP). The method of intelligent fuzzing is that a packet, otherwise correctly constructed, has random values inserted into each of the protocol header fields, with the intent that the packet will be denied when the ruleset is applied. The evaluator ensured a statistically significant sample size, which varied depending on the protocol field length, was used.

The evaluator consulted the TOE audit log, and determined that the TOE was not adversely affected by this testing.

The evaluator used an IPv4 fuzzing tool to direct traffic against the TOE’s management and WAN interfaces, and observed that the TOE successfully dropped all traffic.

The evaluator also performed basic vulnerability analysis of the device in accordance with the requirements of the NDPP. The evaluator used http://www.cvedetails.com to identify known vulnerabilities for each piece of the TOE.

The evaluator searched on cvedetails.com for the following queries:

- IAS
- information assurance specialists
- KG-RU
- router micro

The evaluator was unable to find any vulnerabilities related to the TOE itself. Through the audit log, guidance documentation and the ST, the evaluator determined that the TOE was running the following versions of software: dnsmasq 2.71, and strongswan 5.2.1. The evaluator searched cvedetails.com for vulnerabilities associated with these versions, and listed them below.

The evaluators determined that suitable vulnerabilities would have Low CVSSv2 Access Complexity, because a Medium Access complexity as defined by http://www.first.org/cvss/cvss-guide.html#i2.1.2 requires additional access, social engineering, and/or a non-default configuration.

http://www.cvedetails.com/cve/CVE-2015-3294/ - The evaluator searched exploit-db.com and did not find any publicly available exploits.

9 Results of the Evaluation

The evaluation was carried out in accordance with the Common Criteria Evaluation and Validation Scheme (CCEVS) processes and procedures. The TOE was evaluated against the criteria contained in the Common Criteria for Information Technology Security Evaluation, Version 3.1 Revision 3. The evaluation methodology used by the Evaluation Team to conduct the evaluation is the Common Methodology for Information Technology Security Evaluation, Version 3.1 Revision 3.

InfoGard has determined that the TOE meets the security criteria in the Security Target, which claims compliance with the Protection Profile for Network Devices, Version 1.1, June 8, 2012, the Security Requirements for Network Devices Errata #3, November 3, 2015, and the Network Device Protection Profile Extended Package VPN Gateway, Version 1.1, April 12, 2013.

A team of Validators, on behalf of the CCEVS Validation Body, monitored the evaluation. The evaluation was completed in December 2015.

10 Validator Comments/Recommendations

<TBD>

11 Security Target

IAS Router Security Target, Version 1.0, December 21, 2015.

12 Terms

12.1 Acronyms

CC	Common Criteria
CCIMB	Common Criteria Interpretations Management Board
CCTL	Common Criterial Trusted Lab
CLI	Command Line Interface
CSP	Critical Security Parameters
DAC	Discretionary Access Control
EAL	Evaluation Assurance Level
FIPS	Federal Information Processing Standards Publication 140-2

FTP	File Transfer Protocol
I/O	Input/Output
MIB	Management Information Base
NIST	National Institute of Standards and Technology
OCSP	Online Certificate Status Protocol
PP	Protection Profile
PFE	Packet Forwarding Engine
RE	Routing Engine
SF	Security Functions
SFP	Small Form-factor Pluggable
SFR	Security Functional Requirements
SSH	Secure Shell
SSL	Secure Sockets Layer
ST	Security Target
TOE	Target of Evaluation
TSF	TOE Security Functions

13 Bibliography

[1] Common Criteria for Information Technology Security Evaluation – Part 1: Introduction and general model, dated July 2009, Version 3.1 Revision 3, CCMB-2009-07-001.

[2] Common Criteria (CC) for Information Technology Security Evaluation – Part 2: Security functional components, July 2009, Version 3.1, Revision 3, CCMB-2009-07-002.

[3] Common Criteria for Information Technology Security Evaluation – Part 3: Security assurance components, July 2009, Version 3.1, Revision 3, CCMB-2009-07-003.

[4] Common Methodology for Information Technology Security Evaluation – Evaluation methodology, July 2009, Version 3.1, Revision 3, CCMB-2009-07-004.

14 References

[TRRT2]	E-mail from TRRT	October 3, 2014
[TD13]	TD0013: AVA_VAN.1 in VPN GW EP	September 15, 2014