Supporting Document
Mandatory Technical Document

Evaluation Activities for Stateful Traffic Filter Firewalls cPP

October-2017

Version 2.0

CCDB-2017-<month TBD>-<number TBD>

Foreword

This is a supporting document, intended to complement the Common Criteria version 3 and the associated Common Evaluation Methodology for Information Technology Security Evaluation.

Supporting documents may be “Guidance Documents”, that highlight specific approaches and application of the standard to areas where no mutual recognition of its application is required, and as such, are not of normative nature, or “Mandatory Technical Documents”, whose application is mandatory for evaluations whose scope is covered by that of the supporting document. The usage of the latter class is not only mandatory, but certificates issued as a result of their application are recognized under the CCRA.

This supporting document has been developed by the Network International Technical Community (NDFW-iTC) and is designed to be used to support the evaluations of products against the cPPs identified in section 1.1.

Technical Editor: Network International Technical Community (NDFW-iTC)

Document history:

V2.0, 9 October 2017 (published version)

V1.0, 27 February 2015 (published version)

V0.4, 26 January 2015 (incorporates changes due to comments received from CCDB review)

V0.3, 17 October 2014 (released version following public review, submitted for CCDB review)

V0.2, 13 October 2014 (internal draft in response to public review comments, for iTC review)

V0.1, 5 September 2014 (Initial release for public review)

General Purpose: See section 1.1.

Field of special use: This Supporting Document applies to the evaluation of TOEs claiming conformance with the collaborative Protection Profile for Stateful Traffic Filter Firewalls [FWcPP].

Acknowledgements:

This Supporting Document was developed by the Network international Technical Community with representatives from industry, Government agencies, Common Criteria Test Laboratories, and members of academia.

Table of Contents

1 Introduction

1.1 Technology Area and Scope of Supporting Document

1 This Supporting Document defines the Evaluation Activities associated with the collaborative Protection Profile for Stateful Traffic Filter Firewalls [FWcPP]. Note that [FWcPP] also requires the use of the Evaluation Activities for network devices described in [ND-SD]. This Supporting Document defines only the additional activities for [FWcPP], over and above those in [ND-SD].

2 This Supporting Document is mandatory for evaluations of products that claim conformance to any of the following cPP(s):

a) collaborative Protection Profile for Stateful Traffic Filter Firewalls [FWcPP].

3 Although Evaluation Activities are defined mainly for the evaluators to follow, the definitions in this Supporting Document aim to provide a common understanding for developers, evaluators and users as to what aspects of the TOE are tested in an evaluation against the associated cPPs, and to what depth the testing is carried out. This common understanding in turn contributes to the goal of ensuring that evaluations against the cPP achieve comparable, transparent and repeatable results. In general the definition of Evaluation Activities will also help Developers to prepare for evaluation by identifying specific requirements for their TOE. The specific requirements in Evaluation Activities may in some cases clarify the meaning of SFRs, and may identify particular requirements for the content of Security Targets (especially the TOE Summary Specification), user guidance documentation, and possibly supplementary information (e.g. for entropy analysis or cryptographic key management architecture).

1.2 Structure of the Document

4 Evaluation Activities can be defined for both Security Functional Requirements and Security Assurance Requirements. These are defined in separate sections of this Supporting Document.

5 If any Evaluation Activity cannot be successfully completed in an evaluation then the overall verdict for the evaluation is a ‘fail’. In rare cases there may be acceptable reasons why an Evaluation Activity may be modified or deemed not applicable for a particular TOE, but this must be agreed with the Certification Body for the evaluation.

6 In general, if all Evaluation Activities (for both SFRs and SARs) are successfully completed in an evaluation then it would be expected that the overall verdict for the evaluation is a ‘pass’. To reach a ‘fail’ verdict when the Evaluation Activities have been successfully completed would require a specific justification from the evaluator as to why the Evaluation Activities were not sufficient for that TOE.

7 Similarly, at the more granular level of Assurance Components, if the Evaluation Activities for an Assurance Component and all of its related SFR Evaluation Activities are successfully completed in an evaluation then it would be expected that the verdict for the Assurance Component is a 'pass'. To reach a 'fail' verdict for the Assurance Component when these Evaluation Activities have been successfully completed would require a specific justification from the evaluator as to why the Evaluation Activities were not sufficient for that TOE.

1.3 Application of this Supporting Document

8 This Supporting Document (SD) defines three types of Evaluation Activities (EAs) – TOE Summary Specification (TSS), Guidance Documentation, and Tests and is designed to be used in conjunction with cPPs. cPPs that rely on this SD will explicitly identify it as a source for their EAs[1]. Each security requirement (SFR or SAR) specified in the cPP could have multiple EAs associated with it. The security requirement naming convention is consistent between cPP and SD ensuring a clear one to one correspondence between security requirements and evaluation activities.

9 The cPP and SD are designed to be used in conjunction with each other, where the cPP lists SFRs and SARs and the SD catalogues EAs associated with each SFR and SAR. Some of the SFRs included in the cPP are optional or selection-based. Therefore an ST claiming conformance to the cPP does not necessarily have to include all possible SFRs defined in the cPP.

10 In an ST conformant to the cPP, several operations need to be performed (mainly selections and assignments). Some EAs define separate actions for different selected or assigned values in SFRs. The evaluator shall neither carry out EAs related to SFRs that are not claimed in the ST nor EAs related to specific selected or assigned values that are not claimed in the ST.

11 EAs do not necessarily have to be executed independently from each other. A description in a guidance documentation or one test case, for example, can cover multiple EAs at a time, no matter whether the EAs are related to the same or different SFRs.

1.4 Terminology

1.4.1 Glossary

12 For definitions of standard CC terminology see [CC] part 1.

Term	Meaning
Administrator	See Security Administrator

[1] In general a cPP may reference one or more SDs as sources for the Evaluation Activities for different sets of SFRs.

Term	Meaning
Assurance	Grounds for confidence that a TOE meets the SFRs [CC1].
Required Supplementary Information	Information that is not necessarily included in the Security Target or operational guidance, and that may not necessarily be public. Examples of such information could be entropy analysis, or description of a cryptographic key management architecture used in (or in support of) the TOE. The requirement for any such supplementary information will be identified in the relevant cPP (see description in Section 6).
Security Administrator	The terms "Administrator", "Security Administrator", and "User" are used interchangeably in this document at present and are used to represent a person that has authorized access to the TOE to perform configuration and management tasks.
Target of Evaluation	A set of software, firmware and/or hardware possibly accompanied by guidance. [CC1]
TOE Security Functionality (TSF)	A set consisting of all hardware, software, and firmware of the TOE that must be relied upon for the correct enforcement of the SFRs. [CC1]
TSF Data	Data for the operation of the TSF upon which the enforcement of the requirements relies.
User	See Security Administrator

1.4.2 Acronyms

Acronym	Meaning
cPP	collaborative Protection Profile
CA	Certificate Authority
CN	Certificate Name
CVE	Common Vulnerabilities and Exposures (database)
DN	Domain Name
DNS	Domain Name Service
EA	Evaluation Activity
ECDHE	Elliptic Curve Diffie-Hellman Key Exchange
iTC	International Technical Community
NIST	National Institute of Standards and Technology
SAN	Storage Area Network
SAR	Security Assurance Requirement
SD	Supporting Document
SSL	Secure Sockets Layer
TLS	Transport Layer Security

2 Evaluation Activities for SFRs

13 The EAs presented in this section capture the actions the evaluator performs to address technology specific aspects covering specific SARs (e.g.., ASE_TSS.1, ADV_FSP.1, AGD_OPE.1, and ATE_IND.1) – this is in addition to the CEM work units that are performed in Section 5 (Evaluation Activities for SARs*)*.

14 Regarding design descriptions (designated by the subsections labelled TSS, as well as any required supplementary material that may be treated as proprietary), the evaluator must ensure there is specific information that satisfies the EA. For findings regarding the TSS section, the evaluator's verdicts will be associated with the CEM work unit ASE_TSS.1-1. Evaluator verdicts associated with the supplementary evidence will also be associated with ASE_TSS.1-1, since the requirement to provide such evidence is specified in ASE in the cPP.

15 For ensuring the guidance documentation provides sufficient information for the administrators/users as it pertains to SFRs, the evaluator's verdicts will be associated with CEM work units ADV_FSP.1-7, AGD_OPE.1-4, and AGD_OPE.1-5.

16 Finally, the subsection labelled Tests is where the iTC has determined that testing of the product in the context of the associated SFR is necessary. While the evaluator is expected to develop tests, there may be instances where it is more practical for the developer to construct tests, or where the developer may have existing tests. Therefore, it is acceptable for the evaluator to witness developer-generated tests in lieu of executing the tests. In this case, the evaluator must ensure the developer's tests are executing both in the manner declared by the developer and as mandated by the EA. The CEM work units that are associated with the EAs specified in this section are: ATE_IND.1-3, ATE_IND.1-4, ATE_IND.1-5, ATE_IND.1-6, and ATE_IND.1-7.

Additional Note for Distributed TOEs

17 For a distributed TOE, all examination of Operational Guidance information should be extended to include confirmation that it defines sufficient information to configure individual components such that the overall TOE is correctly established.

18 Evaluation activities for SFRs must be carried out for all distributed TOE components that implement the SFR (as defined in the mapping of SFRs to components – cf. [ND-SD, 5.1.2]). This applies to optional and selection-based SFRs in section 3 and 4 as well as to the core SFRs in this section.

2.1 User Data Protection (FDP)

2.1.1 FDP_RIP.2 Full Residual Information Protection

2.1.1.1 TSS

19 "Resources" in the context of this requirement are network packets being sent through (as opposed to "to", as is the case when a security administrator connects to the TOE) the TOE. The concern is that once a network packet is sent, the buffer or memory area used by the packet still contains data from that packet, and that if that buffer is re-used, those data might remain and make their way into a new packet. The evaluator shall check to ensure that the TSS describes packet processing to the extent that they can determine that no data will be reused when processing network packets. The evaluator shall ensure that this description at a minimum describes how the previous data are zeroized/overwritten, and at what point in the buffer processing this occurs.

2.2 Firewall (FFW)

2.2.1 FFW_RUL_EXT.1 Stateful Traffic Filtering

2.2.1.1 TSS

20 The evaluator shall verify that the TSS provides a description of the TOE's initialization/startup process, which clearly indicates where processing of network packets begins to take place, and provides a discussion that supports the assertion that packets cannot flow during this process.

21 The evaluator shall verify that the TSS also include a narrative that identifies the components (e.g., active entity such as a process or task) involved in processing the network packets and describe the safeguards that would prevent packets flowing through the TOE without applying the ruleset in the event of a component failure. This could include the failure of a component, such as a process being terminated, or a failure within a component, such as memory buffers full and cannot process packets.

2.2.1.2 Guidance Documentation

22 The guidance documentation associated with this requirement is assessed in the subsequent test assurance activities.

2.2.1.3 Tests

23 Test 1: The evaluator shall attempt to get network traffic to flow through the TOE while the TOE is being initialized. A steady flow of network packets that would otherwise be denied by the ruleset should be sourced and be directed at a host. The evaluator shall verify using a packet sniffer that none of the generated network traffic is permitted through the firewall during initialization.

24 Test 2: The evaluator shall attempt to get network traffic to flow through the TOE while the TOE is being initialized. A steady flow of network packets that would be permitted by the ruleset should be sourced and be directed at a host. The evaluator shall verify using a packet sniffer that none of the generated network traffic is permitted through the firewall during initialization and is only permitted once initialization is complete.

25 Note: The remaining testing associated with application of the ruleset is addressed in the subsequent test assurance activities.

FFW_RUL_EXT.1.2/FFW_RUL_EXT.1.3/FFW_RUL_EXT.1.4

TSS

26 The evaluator shall verify that the TSS describes a stateful packet filtering policy and the following attributes are identified as being configurable within stateful traffic filtering rules for the associated protocols:

- ICMPv4
  - Type
  - Code
- ICMPv6
  - Type
  - Code
- IPv4
  - Source address
  - Destination Address
  - Transport Layer Protocol
- IPv6
  - Source address
  - Destination Address
  - Transport Layer Protocol and where defined by the ST author, Extension Header Type, Extension Header Fields
- TCP
  - Source Port
  - Destination Port
- UDP
  - Source Port
  - Destination Port

27 The evaluator shall verify that each rule can identify the following actions: permit or drop with the option to log the operation. The evaluator shall verify that the TSS identifies all interface types subject to the stateful packet filtering policy and explains how rules are associated with distinct network interfaces.

Guidance Documentation

28 The evaluators shall verify that the guidance documentation identifies the following attributes as being configurable within stateful traffic filtering rules for the associated protocols:

- ICMPv4
    - Type
    - Code
- ICMPv6
    - Type
    - Code
- IPv4
    - Source address
    - Destination Address
    - Transport Layer Protocol
- IPv6
    - Source address
    - Destination Address
    - Transport Layer Protocol and where defined by the ST author, Extension Header Type, Extension Header Fields
- TCP
    - Source Port
    - Destination Port
- UDP
    - Source Port
    - Destination Port

29 The evaluator shall verify that the guidance documentation indicates that each rule can identify the following actions: permit, drop, and log.

30 The evaluator shall verify that the guidance documentation explains how rules are associated with distinct network interfaces.

Tests

31 Test 1: The evaluator shall use the instructions in the guidance documentation to test that stateful packet filter firewall rules can be created that permit, drop, and log packets for each of the following attributes:

- ICMPv4
    - Type
    - Code
- ICMPv6
    - Type
    - Code
- IPv4
    - Source address
    - Destination Address
    - Transport Layer Protocol
- IPv6
    - Source address
    - Destination Address
    - Transport Layer Protocol and where defined by the ST author, Extension Header Type, Extension Header Fields

- TCP
  - Source Port
  - Destination Port
- UDP
  - Source Port
  - Destination Port

32 Test 2: Repeat the test assurance activity above to ensure that stateful traffic filtering rules can be defined for each distinct network interface type supported by the TOE.

33 Note that these test activities should be performed in conjunction with those of FFW_RUL_EXT.1.8 where the effectiveness of the rules is tested. The test activities for FFW_RUL_EXT.1.8 define the protocol/attribute combinations required to be tested. If those combinations are configured manually, that will fulfil the objective of these test activities, but if those combinations are configured otherwise (e.g., using automation), these test activities may be necessary in order to ensure the guidance is correct and the full range of configurations can be achieved by a TOE administrator.

FFW_RUL_EXT.1.5

TSS

34 The evaluator shall verify that the TSS identifies the protocols that support stateful session handling. The TSS shall identify TCP, UDP, and ICMP if selected by the ST author.

35 The evaluator shall verify that the TSS describes how stateful sessions are established (including handshake processing) and maintained.

36 The evaluator shall verify that for TCP, the TSS identifies and describes the use of the following attributes in session determination: source and destination addresses, source and destination ports, sequence number, and individual flags.

37 The evaluator shall verify that for UDP, the TSS identifies and describes the following attributes in session determination: source and destination addresses, source and destination ports.

38 The evaluator shall verify that for ICMP (if selected), the TSS identifies and describes the following attributes in session determination: source and destination addresses, other attributes chosen in FFW_RUL_EXT.1.5.

39 The evaluator shall verify that the TSS describes how established stateful sessions are removed. The TSS shall describe how connections are removed for each protocol based on normal completion and/or timeout conditions. The TSS shall also indicate when session removal becomes effective (e.g., before the next packet that might match the session is processed).

Guidance Documentation

40 The evaluator shall verify that the guidance documentation describes stateful session behaviours. For example, a TOE might not log packets that are permitted as part of an existing session.

Tests

41 Test 1: The evaluator shall configure the TOE to permit and log TCP traffic. The evaluator shall initiate a TCP session. While the TCP session is being established, the evaluator shall introduce session establishment packets with incorrect flags to determine that the altered traffic is not accepted as part of the session (i.e., a log event is generated to show the ruleset was applied). After a TCP session is successfully established, the evaluator shall alter each of the session determining attributes (source and destination addresses, source and destination ports, sequence number, flags) one at a time in order to verify that the altered packets are not accepted as part of the established session.

42 Test 2: The evaluator shall terminate the TCP session established per Test 1 as described in the TSS. The evaluator shall then immediately send a packet matching the former session definition in order to ensure it is not forwarded through the TOE without being subject to the ruleset.

43 Test 3: The evaluator shall expire (i.e., reach timeout) the TCP session established per Test 1 as described in the TSS. The evaluator shall then send a packet matching the former session in order to ensure it is not forwarded through the TOE without being subject to the ruleset.

44 Test 4: The evaluator shall configure the TOE to permit and log UDP traffic. The evaluator shall establish a UDP session. Once a UDP session is established, the evaluator shall alter each of the session determining attributes (source and destination addresses, source and destination ports) one at a time in order to verify that the altered packets are not accepted as part of the established session.

45 Test 5: The evaluator shall expire (i.e., reach timeout) the UDP session established per Test 4 as described in the TSS. The evaluator shall then send a packet matching the former session in order to ensure it is not forwarded through the TOE without being subject to the ruleset.

46 Test 6: If ICMP is selected, the evaluator shall configure the TOE to permit and log ICMP traffic. The evaluator shall establish a session for ICMP as defined in the TSS. Once an ICMP session is established, the evaluator shall alter each of the session determining attributes (source and destination addresses, other attributes chosen in FFW_RUL_EXT.1.5) one at a time in order to verify that the altered packets are not accepted as part of the established session.

47 Test 7: If applicable, the evaluator shall terminate the ICMP session established per Test 6 as described in the TSS. The evaluator shall then immediately send a packet matching the former session definition in order to ensure it is not forwarded through the TOE without being subject to the ruleset.

48 Test 8: The evaluator shall expire (i.e., reach timeout) the ICMP session established per Test 6 as described in the TSS. The evaluator shall then send a packet matching the former session in order to ensure it is not forwarded through the TOE without being subject to the ruleset.

FFW_RUL_EXT.1.6

TSS

49 The evaluator shall verify that the TSS identifies the following as packets that will be automatically dropped and are counted or logged:

a) Packets which are invalid fragments, including a description of what constitutes an invalid fragment
b) Fragments that cannot be completely re-assembled
c) Packets where the source address is defined as being on a broadcast network
d) Packets where the source address is defined as being on a multicast network
e) Packets where the source address is defined as being a loopback address
f) The TSF shall reject and be capable of logging network packets where the source or destination address of the network packet is defined as being unspecified (i.e. 0.0.0.0) or an address "reserved for future use" (i.e. 240.0.0.0/4) as specified in RFC 5735 for IPv4;
g) The TSF shall reject and be capable of logging network packets where the source or destination address of the network packet is defined as an "unspecified address" or an address "reserved for future definition and use" (i.e. unicast addresses not in this address range: 2000::/3) as specified in RFC 3513 for IPv6;
h) Packets with the IP options: Loose Source Routing, Strict Source Routing, or Record Route specified
i) Other packets defined in FFW_RUL_EXT.1.6

Guidance Documentation

50 The evaluator shall verify that the guidance documentation describes packets that are discarded and potentially logged by default. If applicable protocols are identified, their descriptions need to be consistent with the TSS. If logging is configurable, the evaluator shall verify that applicable instructions are provided to configure auditing of automatically rejected packets.

Tests

51 Test 1: The evaluator shall test each of the conditions for automatic packet rejection in turn. In each case, the TOE should be configured to allow all network traffic and the evaluator shall generate a packet or packet fragment that is to be rejected. The evaluator shall use packet captures to ensure that the unallowable packet or packet fragment is not passed through the TOE.

52 Test 2: For each of the cases above, the evaluator shall use any applicable guidance to enable dropped packet logging or counting. In each case above,

the evaluator shall ensure that the rejected packet or packet fragment was recorded (either logged or an appropriate counter incremented).

FFW_RUL_EXT.1.7

TSS

53 The evaluator shall verify that the TSS explains how the following traffic can be dropped and counted or logged:

a) Packets where the source address is equal to the address of the network interface where the network packet was received
b) Packets where the source or destination address of the network packet is a link-local address
c) Packets where the source address does not belong to the networks associated with the network interface where the network packet was received, including a description of how the TOE determines whether a source address belongs to a network associated with a given network interface

Guidance Documentation

54 The evaluator shall verify that the guidance documentation describes how the TOE can be configured to implement the required rules. If logging is configurable, the evaluator shall verify that applicable instructions are provided to configure auditing of automatically rejected packets.

Tests

55 Test 1: The evaluator shall configure the TOE to drop and log network traffic where the source address of the packet matches that of the TOE network interface upon which the traffic was received. The evaluator shall generate suitable network traffic to match the configured rule and verify that the traffic is dropped and a log message generated.

56 Test 2: The evaluator shall configure the TOE to drop and log network traffic where the source IP address of the packet fails to match the network reachability information of the interface to which it is targeted, e.g. if the TOE believes that network network 192.168.1.0/24 is reachable through interface 2, network traffic with a source address from the 192.168.1.0/24 network should be generated and sent to an interface other than interface 2. The evaluator shall verify that the network traffic is dropped and a log message generated.

FFW_RUL_EXT.1.8

TSS

57 The evaluator shall verify that the TSS describes the algorithm applied to incoming packets, including the processing of default rules, determination of whether a packet is part of an established session, and application of administrator defined and ordered ruleset.

Guidance Documentation

58 The evaluator shall verify that the guidance documentation describes how the order of stateful traffic filtering rules is determined and provides the necessary instructions so that an administrator can configure the order of rule processing.

Tests

59 Test 1: The evaluator shall devise two equal stateful traffic filtering rules with alternate operations – permit and drop. The rules should then be deployed in two distinct orders and in each case the evaluator shall ensure that the first rule is enforced in both cases by generating applicable packets and using packet capture and logs for confirmation.

60 Test 2: The evaluator shall repeat the procedure above, except that the two rules should be devised where one is a subset of the other (e.g., a specific address vs. a network segment). Again, the evaluator should test both orders to ensure that the first is enforced regardless of the specificity of the rule.

FFW_RUL_EXT.1.9

TSS

61 The evaluator shall verify that the TSS describes the process for applying stateful traffic filtering rules and also that the behavior (either by default, or as configured by the administrator) is to deny packets when there is no rule match unless another required conditions allows the network traffic (i.e., FFW_RUL_EXT.1.5 or FFW_RUL_EXT.2.1).

Guidance Documentation

62 The evaluator shall verify that the guidance documentation describes the behavior if no rules or special conditions apply to the network traffic. If the behavior is configurable, the evaluator shall verify that the guidance documentation provides the appropriate instructions to configure the behavior to deny packets with no matching rules.

Tests

63 For each attribute in FFW_RUL_EXT.1.2, the evaluator shall construct a test to demonstrate that the TOE can correctly compare the attribute from the packet header to the ruleset, and shall demonstrate both the permit and deny for each case. The evaluator shall check the log in each case to confirm that the relevant rule was applied. The evaluator shall record a packet capture for each test to demonstrate the correct TOE behaviour.

FFW_RUL_EXT.1.10

TSS

64 The evaluator shall verify that the TSS describes how the TOE tracks and maintains information relating to the number of half-open TCP connections. The TSS should identify how the TOE behaves when the administratively defined limit is reached and should describe under what circumstances stale half-open connections are removed (e.g. after a timer expires).

Guidance Documentation

65 The evaluator shall verify that the guidance documentation describes the behaviour of imposing TCP half-open connection limits and its default state if unconfigured. The evaluator shall verify that the guidance clearly indicates the conditions under which new connections will be dropped e.g. per-destination or per-client.

Tests

Test 1: The evaluator shall define a TCP half-open connection limit on the TOE. The evaluator shall generate TCP SYN requests to pass through the TOE to the target system using a randomised source IP address and common destination IP address. The number of SYN requests should exceed the TCP half-open threshold defined on the TOE. TCP SYN-ACK messages should not be acknowledged. The evaluator shall verify through packet capture that once the defined TCP half-open threshold has been reached, subsequent TCP SYN packets are not transmitted to the target system. The evaluator shall verify that when the configured threshold is reached that, depending upon the selection, either a log entry is generated or a counter is incremented.

3 Evaluation Activities for Optional Requirements

3.1 Firewall (FFW)

3.1.1 FFW_RUL_EXT.2 Stateful Filtering for Dynamic Protocols

FFW_RUL_EXT.2.1

TSS

66 The evaluator shall verify that the TSS identifies the protocols that can cause the automatic creation of dynamic packet filtering rules. In some cases rather than creating dynamic rules, the TOE might establish stateful sessions to support some identified protocol behaviors.

67 The evaluator shall verify that the TSS explains the dynamic nature of session establishment and removal. The TSS also shall explain any logging ramifications.

68 The evaluator shall verify that for each of the protocols selected, the TSS explains the dynamic nature of session establishment and removal specific to the protocol.

Guidance Documentation

69 The evaluator shall verify that the guidance documentation describes dynamic session establishment capabilities.

70 The evaluator shall verify that the guidance documentation describes the logging of dynamic sessions consistent with the TSS.

Tests

71 Test 1: The evaluator shall define stateful traffic filtering rules to permit and log traffic for each of the supported protocols and drop and log TCP and UDP ports above 1024. Subsequently, the evaluator shall establish a connection for each of the selected protocols in order to ensure that it succeeds. The evaluator shall examine the generated logs to verify they are consistent with the guidance documentation.

72 Test 2: Continuing from Test 1, the evaluator shall determine (e.g., using a packet sniffer) which port above 1024 opened by the control protocol, terminate the connection session, and then verify that TCP or UDP (depending on the protocol selection) packets cannot be sent through the TOE using the same source and destination addresses and ports.

73 Test 3: For each additionally supported protocol, the evaluator shall repeat the procedure above for the protocol. In each case the evaluator must use the applicable RFC or standard in order to determine what range of ports to block in order to ensure the dynamic rules are created and effective.

4 Evaluation Activities for Selection-Based Requirements

74 No additional selection-based requirements are defined in [FWcPP] over and above those defined in [NDcPP].

5 Evaluation Activities for SARs

75 No additional Evaluation Activities for SARs (over and above those in [ND-SD]) are defined here.

76 However, additional details regarding Vulnerability Analysis activities for Stateful Traffic Filter Firewalls are given in Appendix A of the current document.

6 Required Supplementary Information

77 No additional Required Supplementary Information (over and above that in [ND-SD]) is defined here.

7 References

[CC1] Common Criteria for Information Technology Security Evaluation, Part 1: Introduction and General Model
CCMB-2012-09-001, Version 3.1 Revision 4, September 2012

[CC2] Common Criteria for Information Technology Security Evaluation,
Part 2: Security Functional Components,
CCMB-2012-09-002, Version 3.1 Revision 4, September 2012

[CC3] Common Criteria for Information Technology Security Evaluation,
Part 3: Security Assurance Components,
CCMB-2012-09-003, Version 3.1 Revision 4, September 2012

[CEM] Common Methodology for Information Technology Security Evaluation,
Evaluation Methodology,
CCMB-2012-09-004, Version 3.1, Revision 4, September 2012

[FWcPP] collaborative Protection Profile for Stateful Traffic Filter Firewalls,
Version 2.0, 9 October 2017

[NDcPP] collaborative Protection Profile for Network Devices,
Version 2.0, 5 May 2017

[ND-SD] Evaluation Activities for Network Device cPP,
Version 2.0, 5 May 2017

References

Appendices

A. Vulnerability Analysis

78 [ND-SD] contains the details of the vulnerability analysis process to be followed; that information is not repeated here. The additional information that is needed for vulnerability analysis for TOEs conforming to [FWcPP] is contained in the following sections.

A.1 Sources of vulnerability information

79 [ND-SD] identifies the 4 types flaws to be considered by the evaluation team. For each type, the following additional information is provided for TOEs conforming to [FWcPP].

A.1.1 Type 1 Hypotheses – Public-Vulnerability-Based

80 The list of public sources of vulnerability information selected by the iTC is given in Section A.4.

81 The evaluators shall perform a search on the sources listed in Section A.4 to determine a list of potential flaw hypotheses that are more recent that the publication date of the cPP, and those that are specific to the TOE and its components as specified by the additional documentation mentioned above. Any duplicates – either in a specific entry, or in the flaw hypothesis that is generated from an entry from the same or a different source – can be noted and removed from consideration by the evaluation team.

82 The search criteria to be used when searching the sources published after the publication date of the cPP shall include:

- The term “firewall”
- The following protocols: TCP, UDP, IPv4, IPv6
- Any protocols not listed above supported (through an SFR) by the TOE.
- The TOE name (including appropriate model information as appropriate)

83 As part of type 1 flaw hypothesis generation for the specific components of the TOE, the evaluator shall also search the component manufacturer’s websites to determine if flaw hypotheses can be generated on this basis (for instance, if security patches have been released for the version of the component being evaluated, the subject of those patches may form the basis for a flaw hypothesis).

A.1.2 Type 2 Hypotheses – iTC-Sourced

84 Section A.5 contains the list of flaw hypothesis generated by the iTC for this technology that must be considered by the evaluation team as flaw hypotheses in performing the vulnerability assessment.

85 If the evaluators discover a Type 3 or Type 4 flaw that they believe should be considered as a Type 2 flaw in future versions of this cPP, they should work with their Certification Body to determine the appropriate means of submitting the flaw for consideration by the iTC.

A.1.3 Type 3 Hypotheses – Evaluation-Team-Generated

86 Type 3 flaws are formulated by the evaluator based on information presented by the product (through on-line help, product documentation and user guides, etc.) and product behaviour during the (functional) testing activities. The evaluator is also free to formulate flaws that are based on material that is not part of the baseline evidence (e.g., information gleaned from an Internet mailing list, or reading interface documentation on interfaces not included in the set provided by the developer), although such activities have the potential to vary significantly based upon the product and evaluation facility performing the analysis.

87 If the evaluators discover a Type 3 flaw that they believe should be considered as a Type 2 flaw in future versions of this cPP, they should work with their Certification Body to determine the appropriate means of submitting the flaw for consideration by the iTC.

A.1.4 Type 4 Hypotheses – Tool-Generated

88 The evaluator shall perform the following activities to generate type 4 flaw hypotheses:

- Fuzz testing
    - Examine effects of sending:
        - mutated packets carrying each ‘Type’ and ‘Code’ value that is undefined in the relevant RFC for each of ICMPv4 (RFC 792) and ICMPv6 (RFC 4443)
        - mutated packets carrying each ‘Transport Layer Protocol’ value that is undefined in the respective RFC for IPv4 (RFC 791) IPv6 (RFC 2460) should also be covered if it is supported and claimed by the TOE.

      Since none of these packets will match a rule or belong to an allowed session, the packets should not be processed by the TOE, and the TOE should not be adversely affected by this traffic. The evaluator shall ensure the firewall does not allow these packets to flow through the TOE. Any results that are unexpected (e.g., core dumps) are candidates for a flaw hypothesis.
    - Mutation fuzz testing of the remaining fields in the required protocol headers. This testing requires sending mutations of well-formed packets that have both carefully chosen and random values inserted into each header field in turn (i.e. testing is to include both carefully chosen and random insertion test cases). The original well-formed packets would be accepted as part of a normal existing communication stream and may still be accepted as valid packets when subject to the carefully chosen mutations (the individual packet alone would be valid although its contents may not be valid in the context of preceding and/or following

packets), but will often not be valid packets when random values are inserted into fields. The carefully chosen values should include semantically significant values that can be determined from the type of the data that the field represents, such as values indicating positive and negative integers, boundary conditions, invalid binary combinations (e.g. for flag sets with dependencies between bits), and missing start or end values. Randomly chosen values may not result in well-formed packets, but are included nonetheless to see whether they can lead to the device entering an insecure state. Any results that are unexpected (e.g., core dumps) are candidates for a flaw hypothesis.

89 The iTC has not identified a specific tool to be used in accomplishing the above flaw hypothesis generation activity, so any tool used by the evaluation team is acceptable. The evaluation team shall record in the test report the name, version, parameters, and results of all test tools used for this this activity.

90 If the evaluators discover a Type 4 flaw that they believe should be considered as a Type 2 flaw in future versions of this cPP, they should work with their Certification Body to determine the appropriate means of submitting the flaw for consideration by the iTC.

A.2 Process for Evaluator Vulnerability Analysis

91 The process to be followed is described in [ND-SD].

A.3 Reporting

92 Reporting activities are described in [ND-SD].

A.4 Public Vulnerability Sources

93 [ND-SD] identifies the relevant public vulnerability sources to be consulted.

A.5 Additional Flaw Hypotheses

94 No entries are currently defined for this list.

B. Firewall Equivalency Considerations

95 No additional Equivalency Considerations (over and above those in [ND-SD]) are defined here.