Revision Letter for Tsvart Last Call Review of draft-ietf-i2nsf-applicability-13

Editor: Jaehoon Paul Jeong

Date: 07/20/2019

Hi Tommy Pauly,

I sincerely appreciate your valuable comments. I will respond to your comments with my answers starting with the mark “=>”.

Your text uses a bold font and my answer uses a regular font.

OLD is the version 13 and NEW is the version 14:

- OLD: draft-ietf-i2nsf-applicability-13
- NEW: draft-ietf-i2nsf-applicability-14

Reviewer: Tommy Pauly

Review result: Ready with Issues

This document has been reviewed as part of the transport area review team's ongoing effort to review key IETF documents. These comments were written primarily for the transport area directors, but are copied to the document's authors and WG to allow them to address any issues raised and also to the IETF discussion list for information.

When done at the time of IETF Last Call, the authors should consider this review as part of the last-call comments they receive. Please always CC tsv-art@ietf.org if you reply to or forward this review.

Document: draft-ietf-i2nsf-applicability-13

This document provides an overview of how I2NSF can be used in conjunction with firewalls, deep packet inspectors, and DoS mitigation engines. In general, it is concerned with rules for packet-level inspection and interacts with transport protocols on path only by allowing or dropping packets. The impacts that these functions have on transports are not new or specific to this document.

Beyond merely dropping or allowing packets, the document does describe in Section 4 a system that forwards packets between firewalls and web filters. This forwarding refers to RFC 7665 to explain the mechanism for forwarding.

While this document does not present any particular transport-related problems, it does have some clarity and correctness issues. There are also some opportunities for referring to the impact of the firewalling systems on end-to-end transport flows. The issues primarily lie in the example of a time-based firewall system, in Section 4.

=> I addressed transport-related problems such as the security handling of HTTPS traffic in this revision. Also, I addressed the impact of the firewall systems on end-to-end transport flows, especially for a time-based firewall system. All the changes are reflected from OLD to NEW as follows:

- Section 4. Time-dependent Web Access Control Service

OLD:

4. Time-dependent Web Access Control Service

This service scenario assumes that an enterprise network administrator wants to control the staff members' access to a particular Internet service (e.g., Example.com) during business hours. The following is an example high-level security policy rule for a web filter that the administrator requests: Block the staff members' access to Example.com from 9 AM (i.e., 09:00) to 6 PM (i.e., 18:00) by dropping their packets. Figure 2 is an example XML code for this web filter that is sent from the I2NSF User to the Security Controller via the Consumer-Facing Interface [consumer-facing-inf-dm]:

```
<?xml version="1.0" encoding="UTF-8" ?>
<ietf-i2nsf-cfi-policy:policy>
  <policy-name>block_website</policy-name>
  <rule>
    <rule-name>block_website_during_working_hours</rule-name>
    <event>
      <time-information>
        <begin-time>09:00</begin-time>
        <end-time>18:00</end-time>
      </time-information>
    </event>
    <condition>
      <firewall-condition>
        <source-target>
          <src-target>Staff_Member's_PC</src-target>
        </source-target>
      </firewall-condition>
      <custom-condition>
        <destination-target>
          <dest-target>Example.com</dest-target>
        </destination-target>
      </custom-condition>
    </condition>
    <action>
      <primary-action>drop</primary-action>
    </action>
  </rule>
</ietf-i2nsf-cfi-policy:policy>
```

Figure 2: An XML Example for Time-based Web-filter

The security policy name is "block_website" with the tag "policy-name", and the security policy rule name is "block_website_during_working_hours" with the tag "rule-name". The filtering event has the time span where the filtering begin time is the time "09:00" (i.e., 9:00AM) with the tag "begin-time", and the filtering end time is the time "18:00" (i.e., 6:00PM) with the tag "end-time". The filtering condition has the source target of "Staff_Member's_PC" with the tag "src-target", the destination target of a website "Example.com" with the tag "dest-target". The action is to "drop" the packets satisfying the above event and condition with the tag "primary-action".

After receiving the high-level security policy, the Security Controller identifies required security capabilities, e.g., IP address and port number inspection capabilities and URL inspection capability. In this scenario, it is assumed that the IP address and port number inspection capabilities are required to check whether a received packet is an HTTP packet from a staff member. The URL inspection capability is required to check whether the target URL of a received packet is in the Example.com domain or not.

The Security Controller maintains the security capabilities of each active NSF in the I2NSF system, which have been reported by the Developer's Management System via the Registration interface. Based on this information, the Security Controller identifies NSFs that can perform the IP address and port number inspection and URL inspection [policy-translation]. In this scenario, it is assumed that a firewall NSF has the IP address and port number inspection capabilities and a web filter NSF has URL inspection capability.

The Security Controller generates low-level security rules for the NSFs to perform IP address and port number inspection, URL inspection, and time checking. Specifically, the Security Controller may interoperate with an access control server in the enterprise network in order to retrieve the information (e.g., IP address in use, company identifier (ID), and role) of each employee that is currently using the network. Based on the retrieved information, the Security Controller generates low-level security rules to check whether the source IP address of a received packet matches any one being used by a staff member. In addition, the low-level security rules should be able to determine that a received packet is of HTTP protocol. The low-level security rules for web filter check that the target URL field of a received packet is equal to Example.com. Finally, the Security Controller sends the low-level security rules of the IP address and port number inspection to the firewall NSF and the low-level rules for URL inspection to the web filter NSF.

The following describes how the time-dependent web access control service is enforced by the NSFs of firewall and web filter.

1. A staff member tries to access Example.com during business hours, e.g., 10 AM.

2. The packet is forwarded from the staff member's device to the firewall, and the firewall checks the source IP address and port number. Now the firewall identifies the received packet is an HTTP packet from the staff member.

3. The firewall triggers the web filter to further inspect the packet, and the packet is forwarded from the firewall to the web filter. The SFC architecture [RFC7665] can be utilized to support such packet forwarding in the I2NSF framework.

4. The web filter checks the target URL field of the received packet, and realizes the packet is toward Example.com. The web filter then checks that the current time is in business hours. If so, the web filter drops the packet, and consequently the staff member's access to Example.com during business hours is blocked.

NEW:

```
<?xml version="1.0" encoding="UTF-8" ?>
<ietf-i2nsf-cfi-policy:policy>
  <policy-name>block_website</policy-name>
  <rule>
    <rule-name>block_website_during_working_hours</rule-name>
    <event>
      <time-information>
        <begin-time>09:00</begin-time>
        <end-time>18:00</end-time>
      </time-information>
    </event>
    <condition>
      <firewall-condition>
        <source-target>
          <src-target>Staff_Member's_PC</src-target>
        </source-target>
      </firewall-condition>
      <custom-condition>
        <destination-target>
          <dest-target>example.com</dest-target>
        </destination-target>
      </custom-condition>
    </condition>
    <action>
      <primary-action>drop</primary-action>
    </action>
  </rule>
</ietf-i2nsf-cfi-policy:policy>
```

Figure 2: An XML Example for Time-based Web-filter

4. Time-dependent Web Access Control Service

This service scenario assumes that an enterprise network administrator wants to control the staff members' access to a particular Internet service (e.g., example.com) during business hours. The following is an example high-level security policy rule for a web filter that the administrator requests: Block the staff members' access to example.com from 9 AM (i.e., 09:00) to 6 PM (i.e., 18:00) by dropping their packets. Figure 2 is an example XML code for this web filter that is sent from the I2NSF User to the Security Controller via the Consumer-Facing Interface [consumer-facing-inf-dm].

The security policy name is "block_website" with the tag "policy-name", and the security policy rule name is "block_website_during_working_hours" with the tag "rule-name". The filtering event has the time span where the filtering begin time is the time "09:00" (i.e., 9:00AM) with the tag "begin-time", and the filtering end time is the time "18:00" (i.e., 6:00PM) with the tag "end-time". The filtering condition has the source target of "Staff_Member's_PC" with the tag "src-target", and the destination target of a website "example.com" with the tag "dest-target". Note that the destination target can be translated to IP address(es) corresponding to the website's URL, and then either the website's URL or the corresponding IP address(es) can be used by both firewall and web filter. The action is to "drop" the packets satisfying the above event and condition with the tag "primary-action".

After receiving the high-level security policy, the Security Controller identifies required security capabilities, e.g., IP address and port number inspection capabilities and URL inspection capability. In this scenario, it is assumed that the IP address and port number inspection capabilities are required to check whether a received packet is an HTTP-session packet from a staff member, which is part of an HTTP session generated by the staff member. The URL inspection capability is required to check whether the target URL of a received packet is in the example.com domain or not.

The Security Controller maintains the security capabilities of each active NSF in the I2NSF system, which have been reported by the Developer's Management System via the Registration interface. Based on this information, the Security Controller identifies NSFs that can perform the IP address and port number inspection and URL inspection [policy-translation]. In this scenario, it is assumed that a firewall NSF has the IP address and port number inspection capabilities and a web filter NSF has URL inspection capability.

The Security Controller generates low-level security rules for the NSFs to perform IP address and port number inspection, URL inspection, and time checking. Specifically, the Security Controller may interoperate with an access control server in the enterprise network in order to retrieve the information (e.g., IP address in use, company identifier (ID), and role) of each employee that is currently using the network. Based on the retrieved information, the Security Controller generates low-level security rules to check whether the source IP address of a received packet matches any one being used by a staff member.

In addition, the low-level security rules should be able to determine that a received packet uses either the HTTP protocol without Transport Layer Security (TLS) [RFC8446] or the HTTP protocol with TLS as HTTPS. The low-level security rules for web filter check that the target URL field of a received packet is equal to example.com, or that the destination IP address of a received packet is an IP address corresponding to example.com. Note that if HTTPS is used for an HTTP-session packet, the HTTP protocol header is encrypted, so the URL information may not be seen from the packet for the web filtering. Thus, the IP address(es) corresponding to the target URL needs to be obtained from the certificate in TLS versions prior to 1.3 [RFC8446] or the Server Name Indication (SNI) in a TCP-session packet in TLS. Also, to obtain IP address(es) corresponding to a target URL, the DNS name resolution process can be observed through a packet capturing tool because the DNS name resolution will translate the target URL into IP address(es). The IP addresses obtained through either TLS or DNS can be used by both firewall and web filter for whitelisting or blacklisting the TCP five-tuples of HTTP sessions.

Finally, the Security Controller sends the low-level security rules of the IP address and port number inspection to the firewall NSF and the low-level rules for URL inspection to the web filter NSF.

The following describes how the time-dependent web access control service is enforced by the NSFs of firewall and web filter.

1. A staff member tries to access example.com during business hours, e.g., 10 AM.

2. The packet is forwarded from the staff member's device to the firewall, and the firewall checks the source IP address and port number. Now the firewall identifies the received packet is an HTTP-session packet from the staff member.

3. The firewall triggers the web filter to further inspect the packet, and the packet is forwarded from the firewall to the web filter. The SFC architecture [RFC7665] can be utilized to support such packet forwarding in the I2NSF framework.

4. The web filter checks the received packet's target URL field or its destination IP address corresponding to the target URL, and detects that the packet is being sent to the server for example.com. The web filter then checks that the current time is within business hours. If so, the web filter drops the packet, and consequently the staff member's access to example.com during business hours is blocked.

Issues in Section 4:

The text in this example refers several times to an "HTTP packet" as the unit of filtering. This term seems a bit misleading. Presumably, these are packets that comprise a TCP flow (potentially with TLS encryption) on which HTTP messages are being sent. It would be clearer to refer to the packet as being part of an HTTP session or connection. Later, in Section 6, the term "flow of packets" is used; I suggest using that term here as well.

=> Yes, I replaced “an HTTP packet” with “an HTTP-session packet”.

OLD:

In this scenario, it is assumed that the IP address and port number inspection capabilities are required to check whether a received packet is an HTTP packet from a staff member.

NEW:

In this scenario, it is assumed that the IP address and port number inspection capabilities are required to check whether a received packet is an HTTP-session packet from a staff member, which is part of an HTTP session generated by the staff member.

OLD:

Now the firewall identifies the received packet is an HTTP packet from the staff member.

NEW:

Now the firewall identifies the received packet is an HTTP-session packet from the staff member.

By referring to making decisions about "HTTP packets", the text implies that a new decision is being made on a per-packet basis, based on the URL. Any particular packet in a TCP flow being used for HTTP may not contain any URL, but may be a continuation of a message already in progress. To that end, the firewall is almost certainly doing some whitelisting or blacklisting of TCP five-tuples it is already tracking.

=> Yes, I added a text that the firewall performs whitelisting or blacklisting of TCP five-tuples related to the TCP flow for a given HTTP session.

NEW:

Thus, the IP address(es) corresponding to the target URL needs to be obtained from the certificate in TLS versions prior to 1.3 [RFC8446] or the Server Name Indication (SNI) in a TCP-session packet in TLS. Also, to obtain IP address(es) corresponding to a target URL, the DNS name resolution process can be observed through a packet capturing tool because the DNS name resolution will translate the target URL into IP address(es). The IP addresses obtained through either TLS or DNS can be used by both firewall and web filter for whitelisting or blacklisting the TCP five-tuples of HTTP sessions.

There is also no discussion in the example about encrypted traffic. Unless the described system blocks or intercepts all TLS traffic (which I hope it does not), I would assume that a majority of relevant HTTP traffic would be encrypted using TLS. For any such https:// connection, the URL will not be visible to the firewall, and cannot be used for filtering. You may be able to infer the destination using the certificate in TLS versions prior

to 1.3, or the Server Name Indication (SNI) in TLS, but even that may be encrypted.

=>I add the discussion in an example about encrypted traffic such as HTTPS traffic. For a given HTTPS session under a security policy such as firewall or web filter, the URL will not be visible to the firewall or web filter, so it cannot be used for filtering. To figure out the destination IP address corresponding to a URL) under filtering, the destination IP address may be inferred by using the certificate in the Transport Layer Security (TLS), the Service Name Indication (SNI) in TLS, or by observing the DNS name resolution process as follows:

OLD:

> In addition, the low-level security rules should be able to determine that a received packet is of HTTP protocol. The low-level security rules for web filter check that the target URL field of a received packet is equal to Example.com.

NEW:

> In addition, the low-level security rules should be able to determine that a received packet uses either the HTTP protocol without Transport Layer Security (TLS) [RFC8446] or the HTTP protocol with TLS as HTTPS. The low-level security rules for web filter check that the target URL field of a received packet is equal to example.com, or that the destination IP address of a received packet is an IP address corresponding to example.com. Note that if HTTPS is used for an HTTP-session packet, the HTTP protocol header is encrypted, so the URL information may not be seen from the packet for the web filtering. Thus, the IP address(es) corresponding to the target URL needs to be obtained from the certificate in TLS versions prior to 1.3 [RFC8446] or the Server Name Indication (SNI) in a TCP-session packet in TLS. Also, to obtain IP address(es) corresponding to a target URL, the DNS name resolution process can be observed through a packet capturing tool because the DNS name resolution will translate the target URL into IP address(es). The IP addresses obtained through either TLS or DNS can be used by both firewall and web filter for whitelisting or blacklisting the TCP five-tuples of HTTP sessions.

Based on this, I find it surprising that there is little to no discussion of using the remote (or destination) IP address or port of the TCP connection in the firewalling process; rather it discusses using the source/local IP address of the client machine, and the URL. Since the full URL will often not be accessible to a firewall, I would assume that the information about the remote endpoint is often more useful.

=> I added the discussion of using the destination IP address and port number of the TCP connection of the HTTP session related to the URL to be filtered out as follows.

NEW:

> Thus, the IP address(es) corresponding to the target URL needs to be obtained from the certificate in TLS versions prior to 1.3 [RFC8446] or the Server Name Indication (SNI) in a TCP-session packet in TLS. Also, to obtain IP address(es) corresponding to a target URL, the DNS name resolution process can be observed through a packet capturing tool because the DNS name resolution will translate the target URL into IP address(es). The IP addresses obtained through either TLS or DNS can be used by both firewall and web filter for whitelisting or blacklisting the TCP five-tuples of HTTP sessions.

Nits in Section 4:

- I would suggest that you not capitalize "Example.com", but instead refer to "example.com".

=> Yes, I used “example.com” rather than “Example.com”.

- Replace 'current time is in business hours' with 'current time is within business hours', or similar.

=> I used “current time is within business hours” rather than “current time is in business hours”.

- Replace 'realizes the packet is toward Example.com' with 'detects that the packet is being sent to the server for "example.com"', or similar.

=> I used “detects that the packet is being sent to the server for example.com” rather than “realizes the packet is toward Example.com”.

- It is unclear what types are allows for "dest-target"; the given string is a hostname, but much of the text refers to URLs.

=> “dest-target” is a URL that can be translated into an IP address (or IP addresses). The web filter can use the URL or its translated IP address(es) for web filtering.

OLD:

The filtering condition has the source target of "Staff_Member's_PC" with the tag "src-target", the destination target of a website "Example.com" with the tag "dest-target".

NEW:

The filtering condition has the source target of "Staff_Member's_PC" with the tag "src-target", and the destination target of a website "example.com" with the tag "dest-target". Note that the destination target can be translated to IP address(es) corresponding to the website’s URL, and then either the website’s URL or the corresponding IP address(es) can be used by both firewall and web filter.

Thanks.

Best Regards,

Jaehoon Paul Jeong