

Check Point Configuration with Radware (Alteon) SSL Decrypt with URL/UserCheck Functionality

Mike Walsh
 Security Engineer, US Central
 December 21, 2018

Contents

- Check Point/Radware Design 1
- Check Point URL Filtering 1
- Problem Description/Troubleshooting 2&3
- Kernel Parameter Workaround and Hotfix 4
- Radware URL Rule..... 5

Check Point/Radware Design Goal

The goal of this design was to facilitate the use of Radware/Alteon devices to decrypt SSL traffic and redirect it to Check Point for advanced inspection. This inspection included the use of Check Point URL filtering which was configured to provide a UserCheck page to the end user. The flow is described as follows:

Client -> Alteon (HTTPS) 443 -> Alteon (HTTP) 8080 -> Check Point FW (HTTP) 8080 -> Alteon (HTTP) 8080 -> Alteon (HTTPS) 443 -> Server 443

Check Point URL Filtering

Check Point has included web filtering with its security gateways for many years. URL Filtering is a standard blade included with all Next Generation Threat Prevention (NGTP) and Next Generation Threat Extraction (NGTX) packages. As is common with most major web filtering vendors, Check Point tracks over 200 million websites and categorizes them into 70+ categories for easy identification and policy configuration. These categories are typically used in the Access Control policy to Block and/or Allow websites by their category tags, with specific sites being identified by the creation of Custom Site to allow for policy exceptions to the general category, or for categorization override. An example URLF policy for Check Point R80.10 is shown here:

No.	Name	Source	Destination	Services & Applications	Content	Action	Track
5	Access to Internet according to Web control policy	InternalZone	Internet	* Any	* Any	Web Control	N/A
5.1	Block abuser/ high risk applications	Corporate LANs Branch Office LAN	Internet	Child Abuse Gambling High Risk Pornography Spyware / Malicious Sites	* Any	Drop Blocked Message	Log
5.2	HR can access to social network applications	HR	Internet	Social Networking	* Any	Inform Access Approval Once a day Per application/site	Log Accounting
5.3	All employees can access Youtube for work purposes	Corporate LANs Branch Office LAN	Internet	YouTube - Custom Site	* Any	Ask Company Policy Once a day Per application/site	Log
5.4	Block specific URLs	* Any	Internet	Blocked URLs	* Any	Drop	Log
5.5	Block specific categories for all employees	Corporate LANs Branch Office LAN	Internet	Social Networking Streaming Media Protocols P2P File Sharing	* Any	Drop Blocked Message	Log
5.6	Cleanup	* Any	* Any	* Any	* Any	Accept	Log

Problem Description

In testing the design in our customers environment, we came across an issue where traffic that was sent from the end user through Alteon A to the Check Point firewall returned a "This site can't be reached" page to the end user as opposed to the configured UserCheck block page.



This site can't be reached

The connection was reset.

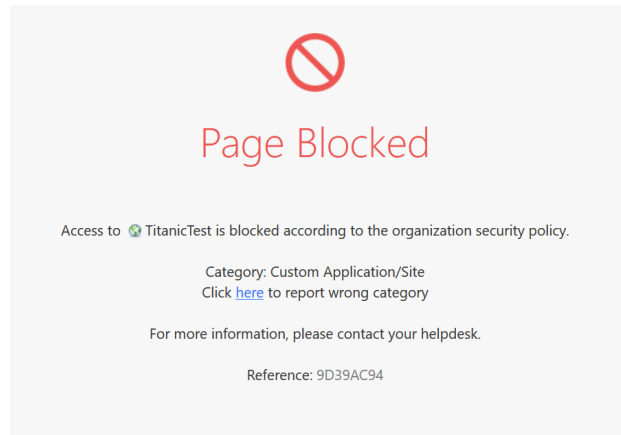
Try:

- Checking the connection
- Checking the proxy and the firewall
- Running Windows Network Diagnostics

ERR_CONNECTION_RESET

Reload

Details



Troubleshooting/Problem Determination

Working with the customer and Radware we took the following Wireshark Capture and provided this analysis.

1. Three way handshake is completed, we can see all four stages on the CP gateway ('I', 'o' and 'O') from the internal interface eth1 to the external interface eth2 and vice versa:

No.	Time	Source	Destination	Protocol	SrcPort	DstPort	Length	CP	Info
1853	2018-09-06 13:16:21.184647	10.95.26.125	192.229.173.125	TCP	59111	8443	78	eth2 I eth1	59111 → 8443 [SYN] Seq=0 Win=65535 Len=0 MSS=1460 IS=0 SACK_PERM=1 TSval=1380736924 TSecr=0
1854	2018-09-06 13:16:21.185135	10.95.26.125	192.229.173.125	TCP	59111	8443	78	eth2 I eth1	[TCP Out-Of-Order] 59111 → 8443 [SYN] Seq=0 Win=65535 Len=0 MSS=1460 IS=0 SACK_PERM=1 TSval=1380736924 TSecr=0
1855	2018-09-06 13:16:21.185211	10.95.26.125	192.229.173.125	TCP	59111	8443	78	eth2 o eth1	[TCP Out-Of-Order] 59111 → 8443 [SYN] Seq=0 Win=65535 Len=0 MSS=1460 IS=0 SACK_PERM=1 TSval=1380736924 TSecr=0
1856	2018-09-06 13:16:21.185273	200.71.0.81	192.229.173.125	TCP	10400	8443	78	O eth2	eth1 10400 → 8443 [SYN] Seq=0 Win=65535 Len=0 MSS=1460 IS=0 SACK_PERM=1 TSval=1380736924 TSecr=0
1857	2018-09-06 13:16:21.185383	192.229.173.125	200.71.0.81	TCP	8443	10400	78	I eth2	eth1 8443 → 10400 [SYN, ACK] Seq=0 Ack=1 Win=65535 Len=0 MSS=1460 IS=0 SACK_PERM=1 TSval=1380736924 TSecr=1380736924
1858	2018-09-06 13:16:21.185416	192.229.173.125	10.95.26.125	TCP	8443	59111	78	eth2 I eth1	8443 → 59111 [SYN, ACK] Seq=0 Ack=1 Win=65535 Len=0 MSS=1460 IS=0 SACK_PERM=1 TSval=1380736924 TSecr=1380736924
1859	2018-09-06 13:16:21.185443	192.229.173.125	10.95.26.125	TCP	8443	59111	78	eth2 o eth1	[TCP Out-Of-Order] 8443 → 59111 [SYN, ACK] Seq=0 Ack=1 Win=65535 Len=0 MSS=1460 IS=0 SACK_PERM=1 TSval=1380736924 TSecr=1380736924
1860	2018-09-06 13:16:21.185500	192.229.173.125	192.229.173.125	TCP	8443	59111	78	eth2 O eth1	[TCP Out-Of-Order] 8443 → 59111 [SYN, ACK] Seq=0 Ack=1 Win=65535 Len=0 MSS=1460 IS=0 SACK_PERM=1 TSval=1380736924 TSecr=1380736924
1861	2018-09-06 13:16:21.185585	10.95.26.125	192.229.173.125	TCP	59111	8443	66	eth2 I eth1	59111 → 8443 [ACK] Seq=1 Ack=1 Win=263536 Len=0 TSval=1380736924 TSecr=1380736924
1862	2018-09-06 13:16:21.185914	10.95.26.125	192.229.173.125	TCP	59111	8443	66	eth2 o eth1	[TCP Dup ACK 100101] 59111 → 8443 [ACK] Seq=1 Ack=1 Win=263536 Len=0 TSval=1380736924 TSecr=1380736924
1863	2018-09-06 13:16:21.185970	10.95.26.125	192.229.173.125	TCP	59111	8443	66	eth2 o eth1	[TCP Dup ACK 100102] 59111 → 8443 [ACK] Seq=1 Ack=1 Win=263536 Len=0 TSval=1380736924 TSecr=1380736924
1864	2018-09-06 13:16:21.185993	200.71.0.81	192.229.173.125	TCP	10400	8443	66	O eth2	eth1 10400 → 8443 [ACK] Seq=1 Ack=1 Win=263536 Len=0 TSval=1380736924 TSecr=1380736924

2. HEAD, ACK and HTTP 200 OK packets pass through the gateway:

1865	2018-09-06 13:16:21.186915	10.95.26.125	192.229.173.125	HTTP	59111	8443	271	eth2 I eth1	HEAD /rdurfsckfscfck HTTP/1.1
1866	2018-09-06 13:16:21.186918	10.95.26.125	192.229.173.125	TCP	59111	8443	271	eth2 eth1	[TCP Retransmission] 59111 → 8443 [PSH, ACK] Seq=1 Ack=1 Win=263536 Len=0 TSval=1380736924 TSecr=1380736924
1867	2018-09-06 13:16:21.186966	10.95.26.125	192.229.173.125	TCP	59111	8443	271	eth2 o eth1	[TCP Retransmission] 59111 → 8443 [PSH, ACK] Seq=1 Ack=1 Win=263536 Len=0 TSval=1380736924 TSecr=1380736924
1868	2018-09-06 13:16:21.186995	200.71.0.81	192.229.173.125	HTTP	10400	8443	271	O eth2	HEAD /rdurfsckfscfck HTTP/1.1
1869	2018-09-06 13:16:21.186980	192.229.173.125	200.71.0.81	TCP	8443	10400	66	I eth2	eth1 8443 → 10400 [ACK] Seq=1 Ack=206 Win=263536 Len=0 TSval=1380736924 TSecr=1380736924
1870	2018-09-06 13:16:21.186986	192.229.173.125	10.95.26.125	TCP	8443	59111	66	eth2 I eth1	8443 → 59111 [ACK] Seq=1 Ack=206 Win=263536 Len=0 TSval=1380736924 TSecr=1380736924
1871	2018-09-06 13:16:21.186975	192.229.173.125	10.95.26.125	TCP	8443	59111	66	eth2 eth1	[TCP Dup ACK 107041] 8443 → 59111 [ACK] Seq=1 Ack=206 Win=263536 Len=0 TSval=1380736924 TSecr=1380736924
1872	2018-09-06 13:16:21.186986	192.229.173.125	10.95.26.125	TCP	8443	59111	66	eth2 O eth1	[TCP Dup ACK 107042] 8443 → 59111 [ACK] Seq=1 Ack=206 Win=263536 Len=0 TSval=1380736924 TSecr=1380736924
1873	2018-09-06 13:16:21.186918	192.229.173.125	200.71.0.81	HTTP	8443	10400	109	I eth2	HTTP/1.1 200 OK
1874	2018-09-06 13:16:21.186942	192.229.173.125	10.95.26.125	HTTP	8443	59111	109	eth2 I eth1	HTTP/1.1 200 OK
1875	2018-09-06 13:16:21.186955	192.229.173.125	10.95.26.125	HTTP	8443	59111	109	eth2 eth1	[TCP Fast Retransmission] HTTP/1.1 200 OK
1876	2018-09-06 13:16:21.187011	192.229.173.125	10.95.26.125	HTTP	8443	59111	109	eth2 O eth1	[TCP Fast Retransmission] HTTP/1.1 200 OK

3. However when the gateway gets the GET packet it doesn't pass through, instead the gateway sends an ACK to the website:

1877	2018-09-06 13:16:21.191247	10.95.26.125	192.229.173.125	HTTP	59111	8443	534	eth2	i	eth1	GET / HTTP/1.1
1878	2018-09-06 13:16:21.191271	10.95.26.125	192.229.173.125	HTTP	59111	8443	534	eth2	i	eth1	[TCP Fast Retransmission] GET / HTTP/1.1
1879	2018-09-06 13:16:21.191765	10.95.26.125	192.229.173.125	TCP	59111	8443	54	eth2	o	eth1	59111 → 8443 [ACK] Seq=206 Ack=44 Win=198896 Len=0
1880	2018-09-06 13:16:21.191798	208.71.0.81	192.229.173.125	TCP	10400	8443	54	eth2	o	eth1	10400 → 8443 [ACK] Seq=206 Ack=44 Win=198896 Len=0

4. Looking at the debugs I can see that the CP gateway is stopping the connection and trying to inject the block page. I also see that it vanishes the original packet and sends ACK to the server instead:

```

; 6Sep2018 17:16:21.191635:[cpu_6];[fw4_2];psl_set_server_injection_ex: setting injection to connection <dir 1, 10.95.26.125:59111 ->
192.229.173.125:8443 IPP 6>;
; 6Sep2018 17:16:21.191637:[cpu_6];[fw4_2];psl_set_reject_conn: added reject dirs 1 to astream=ffffc20091226570 (flags 0x40448c1/0xd27/0xd07);
; 6Sep2018 17:16:21.191639:[cpu_6];[fw4_2];psl_process_data: processing function for app 3[HTTP_DISPATCHER] returned STOP_AND_INJECT;
; 6Sep2018 17:16:21.191706:[cpu_6];[fw4_2];psl_handle_segment_injection_top: vanishing original packet and sending stripped (ack-only) packet.;

```

5. It looks like it timed out waiting for the internal server to launch the block page:

```

; 6Sep2018 17:16:22.231583:[cpu_6];[fw4_2];psl_update_segment_injection_cb: server idle timeout reached at time 1536254182;

```

6. Then after a second the client PC (or Radware device) sends a retransmission packet of the GET command (PSH,ACK) and right after the gateway sends RST packets to both client and server:

1964	2018-09-06 13:16:22.234292	10.95.26.125	192.229.173.125	TCP	59111	8443	534	eth2	i	eth1	[TCP Retransmission] 59111 → 8443 [PSH, ACK] Seq=206 Ack=44 Win=263488 Len=408 TSecr=138877974 TSecr=138877974
1966	2018-09-06 13:16:22.234355	10.95.26.125	192.229.173.125	TCP	59111	8443	54	eth2	o	eth1	59111 → 8443 [RST] Seq=674 Win=198896 Len=0
1967	2018-09-06 13:16:22.234381	208.71.0.81	192.229.173.125	TCP	10400	8443	54	eth2	o	eth1	10400 → 8443 [RST] Seq=674 Win=198896 Len=0
1968	2018-09-06 13:16:22.234410	192.229.173.125	10.95.26.125	TCP	8443	59111	54	eth2	o	eth1	8443 → 59111 [RST] Seq=44 Win=198896 Len=0
1969	2018-09-06 13:16:22.234432	192.229.173.125	10.95.26.125	TCP	8443	59111	54	eth2	o	eth1	8443 → 59111 [RST] Seq=44 Win=198896 Len=0

7. When the gateway sees the retransmission, it drops the packet and attempts to inject the block page again, however aborts because it was already done and therefore sends the RST packets to both client and server:

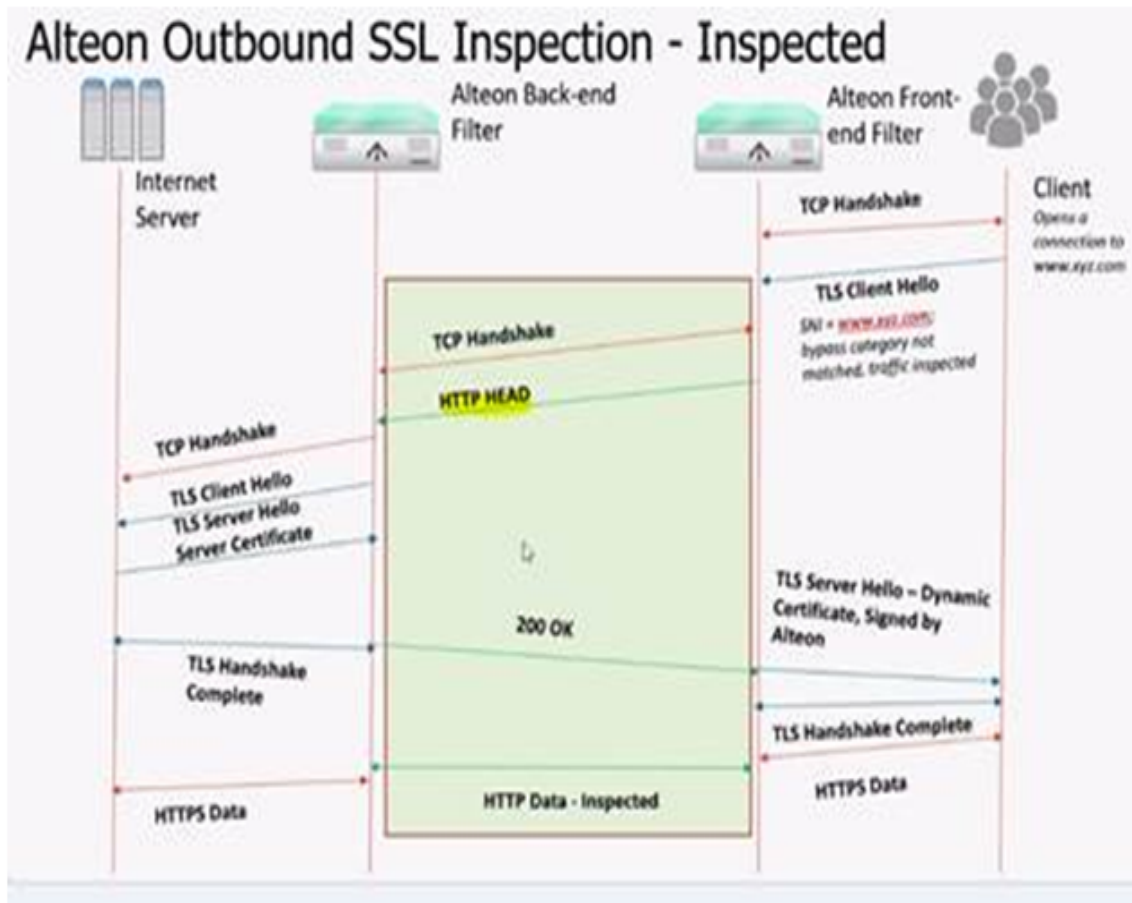
```

; 6Sep2018 17:16:22.231585:[cpu_6];[fw4_2];psl_can_send_injection_ex: already passed packets on injected direction, aborting injection;
; 6Sep2018 17:16:22.231585:[cpu_6];[fw4_2];psl_verify_can_send_injection: sending RST to client and server;

```

In working with Check Point R&D it was determined that the Gateway was behaving as designed and that the issue matched the below SK. The issue relates to the requirement that Radware has in the order at which it sends HEAD requests between Alteon devices. This requirement is proprietary to Radware. The customer was also testing A10 and F5 and we did not have this issue.

SK108312 - When browsing to a page blocked by Application Control policy, the block page is displayed only occasionally



Since this packet is passed first (not expected in regular HTTP traffic), then when the CP gateway sees the GET packet it attempts to inject the UserCheck page but aborts because the connection already passed the HEAD packet:

```
; 6Sep2018 17:16:22.231585;[cpu_6];[fw4_2];psl_can_send_injection_ex: already passed packets on injected direction, aborting injection;
```

Kernel Parameter Workaround

Per SK108312 there is a kernel parameter that can be set to allow the gateway to accept the order of packets required by Radware (command below). However, this opens a security vulnerability on the gateway. Check Point recommends usage of the kernel parameter for testing only.

```
fw ctl set int psl_verify_segment_injection = 0
```

Check Point Hotfix

Working with Check Point R&D, they produced the following Hotfix - HOTFIX_R80_10_JHF_142_281 to be installed on top of JHF142 for R80.10 which the customer has deployed. This patch, while it does not set the above kernel parameter it does allow the gateway to handle specifically an out of order HTTP HEAD request and close the security vulnerability.

Radware URL Rule

Additionally, based on the gateway policy the HTTP HEAD was being blocked, this needs to pass through, the next request which is the actual client request needs to be redirected by the user check.

HEAD /internalheadreq HTTP/1.1 ← This HEAD method needs to be allowed always.

Connection: Keep-Alive

Host: radware.com

Server_name: radware.com

Server_port: 443

App_id: 4

Session_id: 25300

Certificate: Request

So, we created a URL Custom Site to allow the communication.

No.	Name	Source	Destination	VPN	Services & Applications	Action	Track	Install On
1	SSLDE TESTING	HST-LAB_10.95.26.125	Internet	* Any	internalheadreq	Accept	Log Accounting	* Policy Targets

*/internalheadreq – this is the URL that needs to be allowed.

With these pieces in place we have a working solution with the Radware (Alteon SSL Offload) with URL filtering with UserCheck.