Popular open source shopping cart app Zen Cart is warning its users of dozens of cross-site scripting vulnerabilities found in its software. Affected websites, security experts say, risk exposing customers to malware, theft of cookies data and site defacement.

Zen Cart, with an estimated 113,000 active users (according to BuiltWith.com), has told its users they will have to pro-actively install the patch. Affected customers told Threatpost that Zen Cart has notified them of the vulnerability offering

For its part, Trustwave told Threatpost that 50 XSS vulnerabilities were found in the admin section of the Zen Cart software along with one issue in the non-authentication portion of the application.

“We discovered the XSS on Zen Cart in a completely random manner,” said Alex Rothacker, senior security researcher at Trustwave. “We have a monthly Hack Friday event where the team goes ahead and picks something and tries to find vulnerabilities. In this case, one team member picked Zen Cart, because it was a popular solution.”
network security

cs642
adam everspaugh
computer security
ace@cs.wisc.edu
today

* Crypto exercise in-class
* Link layer (in-)security
* IP, TCP (in-)security
crypto exercise
Local area network (LAN)

Ethernet

802.11

Internet backbone

ISP1

ISP2

Alice

Bob

Internet

TCP/IP

BGP (border gateway protocol)

DNS (domain name system)
## Internet protocol stack

<table>
<thead>
<tr>
<th>Layer</th>
<th>Protocols</th>
</tr>
</thead>
<tbody>
<tr>
<td>Application</td>
<td>HTTP, FTP, SMTP, SSH, etc.</td>
</tr>
<tr>
<td>Transport</td>
<td>TCP, UDP</td>
</tr>
<tr>
<td>Network</td>
<td>IP, ICMP, IGMP</td>
</tr>
<tr>
<td>Link</td>
<td>802x (802.11, Ethernet)</td>
</tr>
</tbody>
</table>
Internet protocol stack

Application
TCP
IP
Ethernet

user data
Appl hdr
user data
TCP hdr
user data
IP hdr
TCP hdr
Appl hdr
user data
ENet hdr
IP hdr
TCP hdr
Appl hdr
user data
ENet tlrl

14
20
20

46 to 1500 bytes
IP routing:
Figure out where to send an IP packet based on destination address.

Link layer and IP must cooperate to route packets

ARP enables this cooperation by mapping IPs to MACs
ARP caches

• Hosts maintain cache of ARP data
  – just a table mapping between IPs and MACs
ARP has no authentication

- Easy to sniff packets on (non-switched) ethernet
- What else can we do?

**Easy Denial of Service (DoS):**
Send ARP reply associating gateway 192.168.1.1 with a non-used MAC address
ARP has no authentication

- Easy to sniff packets on (non-switched) ethernet
- What else can we do?

Active Man-in-the-Middle:

ARP reply to MAC2
192.168.1.1 -> MAC3

ARP reply to MAC1
192.168.1.2 -> MAC3

Now traffic “routed” through malicious box
802.11 (wifi)

STA = station
AP = access point

BSS = basic service set
DS = distribution service
ESS = extended service set

SSID (service set identifier) identifies the 802.11 network

Typical WiFi modes:
Unsecured
Wireless Protected Access (WPA2) - password authenticated, encrypted

802.11 association

Probe request
SSID: “linksys”, BSSID: MAC1
Auth request MAC1
Auth response
Associate request MAC1
Associate response
802.11 association

Two APs for same network

AP1

AP2

Probe request
SSID: “linksys”, BSSID: MAC1
SSID: “linksys”, BSSID: MAC2

Auth request MAC2

Choose one of MAC1, MAC2

MAC1

MAC2
802.11 evil twins

Basic idea:
- Attacker pretends to be an AP to intercept traffic or collect data

Basic attack: rogue AP

Choose one of MAC1, MAC2

Probe request
SSID: “linksys”, BSSID: MAC1
SSID: “linksys”, BSSID: MAC2

Auth request MAC2

...
Parrot ARdrone

Drone is a WiFi access point
Uses unsecured 802.11 connection (WiFi)
Controlled from iPad or iPhone with an app
Uses MAC address for security
Internet protocol stack

Application
TCP
IP
Ethernet

TCP segment
IP datagram
Ethernet frame

46 to 1500 bytes
IP protocol (IPv4)

• Connectionless
  – no state

• Unreliable
  – no guarantees

• ICMP (Internet Control Message Protocol)
  – error messages, etc.
  – often used by tools such as ping, traceroute
### IPv4

An Ethernet frame containing an IP datagram is structured as follows:

<table>
<thead>
<tr>
<th>4-bit version</th>
<th>4-bit hdr len</th>
<th>8-bit type of service</th>
<th>16-bit total length (in bytes)</th>
</tr>
</thead>
<tbody>
<tr>
<td>16-bit identification</td>
<td>3-bit flags</td>
<td>13-bit fragmentation offset</td>
<td></td>
</tr>
<tr>
<td>8-bit time to live (TTL)</td>
<td>8-bit protocol</td>
<td>16-bit header checksum</td>
<td></td>
</tr>
<tr>
<td>32-bit source IP address</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>32-bit destination IP address</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>options (optional)</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Security issues with IP

Routing has issues, we’ll get to that later
What else?
- No source address authentication in general
Denial of Service (DoS) attacks

Goal: prevent legitimate users from accessing victim (1.2.3.4)

ICMP ping flood
## ICMP

(Internet Control Message Protocol)

### Diagram

- **IP hdr**
- **ICMPHdr**
- **ICMP message**

### Structure

<table>
<thead>
<tr>
<th>8-bit type</th>
<th>8-bit code</th>
<th>16-bit checksum</th>
</tr>
</thead>
<tbody>
<tr>
<td>4-byte more of header (depends on type)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>message ...</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Denial of Service (DoS) attacks

Goal is to prevent legitimate users from accessing victim (1.2.3.4)

ICMP ping flood
- Attacker sends ICMP pings as fast as possible to victim
- When will this work as a DoS? Attacker resources > victim’s
- How can this be prevented? Ingress filtering near victim
Denial of Service (DoS) attacks

How can attacker avoid ingress filtering?

Attacker can send packet with fake source IP “spoofed” packet
Packet will get routed correctly
Replies will not

Send IP packet with
source: 8.7.3.4
dest: 1.2.3.4
from 5.6.7.8

Filter based on source may be incorrect
DoS reflection attacks

Note a valid packet sends a reply to 8.7.3.4
- Attacker can bounce an attack against 8.7.3.4 off 1.2.3.4
- “Frame” 1.2.3.4
- Single-packet exploit (1.2.3.4 in foreign country)
Denial of Service (DoS) attacks

DoS works better when there is *asymmetry* between victim and attacker
- Attacker uses few resources to cause victim to consume lots of resources
Denial of Service (DoS) attacks

DoS works better when there is **asymmetry** between victim and attacker
- Attacker uses few resources to cause victim to consume lots of resources

Old example: Smurf attack
Router allows attacker to send broadcast ICMP ping on network. Attacker spoofs SRC address to be 1.2.3.4
Denial of Service (DoS) attacks

DoS works better when there is **asymmetry** between victim and attacker
- Attacker uses few resources to cause victim to consume lots of resources

More recent: DNS reflection attacks
Send DNS request w/ spoofed target IP (~65 byte request)
DNS replies sent to target (~512 byte response)
* In-class exercise
  / Hybrid encryption, digital signatures, PBKDF

* Network Security
  / ARP cache poisoning, MitM, DoS
  / WiFi Evil Twins
  / IP (in-security)

* Exit slips
  / 1 thing you learned
  / 1 thing you didn't understand