Infected ATMs Give Away Millions of Dollars Without Credit Cards

Soulskill posted yesterday | from the i'll-order-a-dozen dept.

An anonymous reader writes:

Kaspersky Lab performed a forensic investigation into cybercriminal attacks targeting multiple ATMs around the world. During the course of this investigation, researchers discovered the Tyupkin malware used to infect ATMs and allow attackers to remove money via direct manipulation, stealing millions of dollars. The criminals work in two stages. First, they gain physical access to the ATMs and insert a bootable CD to install the Tyupkin malware. After they reboot the system, the infected ATM is now under their control and the malware runs in an infinite loop waiting for a command. To make the scam harder to spot, the Tyupkin malware only accepts commands at specific times on Sunday and Monday nights. During those hours, the attackers are able to steal money from the infected machine.

US Says It Can Hack Foreign Servers Without Warrants

Soulskill posted 6 hours ago | from the how-to-win-friends dept.

Advocatus Diaboli tips news that the U.S. government is now arguing it doesn't need warrants to hack servers hosted on foreign soil. At issue is the current court case against Silk Road operator Ross Ulbricht. We recently discussed how the FBI's account of how they obtained evidence from Silk Road servers didn't seem to mesh with reality. Now, government lawyers have responded in a new court filing (PDF). They say that even if the FBI had to hack those servers without a warrant, it doesn't matter, because the Fourth Amendment does not confer protection...
DNS and BGP

CS642: Computer Security

Professor Ristenpart

http://www.cs.wisc.edu/~rist/

rist at cs dot wisc dot edu
DNS and BGP
We don’t want to have to remember IP addresses.

Early days of ARPANET: manually managed hosts.txt served from single computer at SRI.
Heirarchical domain name space

(root) org net edu com tv ca

wisc ucsd davis

cs ece

ICANN (Internet Corporation for Assigned Names and Numbers)

root nameservers and authoritative nameservers

max 63 characters

Zone: subtree
Resolving names

From
Example DNS query types

<table>
<thead>
<tr>
<th>Query Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>address (get me an IPv4 address)</td>
</tr>
<tr>
<td>AAAA</td>
<td>IPv6 address</td>
</tr>
<tr>
<td>NS</td>
<td>name server</td>
</tr>
<tr>
<td>TXT</td>
<td>human readable text, has been used for some encryption mechanisms</td>
</tr>
<tr>
<td>MX</td>
<td>mail exchange</td>
</tr>
</tbody>
</table>
Caching

• DNS servers will cache responses
  – Both negative and positive responses
  – Speeds up queries
  – periodically times out. TTL set by data owner
DNS packet on wire

Query ID is 16-bit random value

We’ll walk through the example from Friedl’s document

From Friedl explanation of DNS cache poisoning, as are following diagrams
Query from resolver to NS

<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>src IP</td>
<td>68.94.156.1</td>
</tr>
<tr>
<td>dst IP</td>
<td>192.26.92.30</td>
</tr>
<tr>
<td>src port</td>
<td>5798</td>
</tr>
<tr>
<td>dst port</td>
<td>53</td>
</tr>
<tr>
<td>QID</td>
<td>43561</td>
</tr>
<tr>
<td>Qu</td>
<td>What is A record for <a href="http://www.unixwiz.net">www.unixwiz.net</a>?</td>
</tr>
</tbody>
</table>

- **RD=1** - recursion desired
- **OP=0** - standard query
- **QR=0** - this is a query

- dnsr1.sbcglobal.net
- c.gtld-servers.net
Reply from NS to Resolver

<table>
<thead>
<tr>
<th>IP</th>
<th>TTL</th>
<th>protocol</th>
<th>header cksum</th>
</tr>
</thead>
<tbody>
<tr>
<td>src IP = 192.26.92.30</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>dst IP = 68.94.156.1</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>UDP</th>
<th>src port = 53</th>
<th>dst port = 5798</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>UDP length</td>
<td></td>
</tr>
<tr>
<td></td>
<td>UDP cksum</td>
<td></td>
</tr>
<tr>
<td>QID = 43561</td>
<td>Operation = 0</td>
<td>Query ID Count = 1</td>
</tr>
<tr>
<td></td>
<td>Additional = 1</td>
<td>Question Count = 1</td>
</tr>
<tr>
<td>Authority count = 2</td>
<td>Answer Count = 0</td>
<td></td>
</tr>
<tr>
<td>Addl. Record count = 2</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Query: What is A record for www.unixwiz.net?

Mandatory:
- Unixwiz.net NS = linux.unixwiz.net 2 dy
- Unixwiz.net NS = cs.unixwiz.net 2 dy
- Linux.unixwiz.net A = 64.170.162.98 1 hr
- Cs.unixwiz.net A = 8.7.25.94 1 hr

Optional:
- Glue Records
- TTL

Response contains IP addr of next NS server (called “glue”)

Response ignored if unrecognized QueryID
Query to Second NS

![Diagram showing query and responses for DNS]

- **src IP**: 68.94.156.1
- **dst IP**: 64.170.162.98
- **src port**: 5798
- **dst port**: 53
- **QID**: 43562
- **Question count**: 1
- **Answer count**: 0
- **Authority count**: 0
- **Additional Record count**: 0

**Query:** What is A record for www.unixwiz.net?

**DNS Responses:**
- dnsr1.sbcglobal.net
- linux.unixwiz.net

**Notes:**
- RD=1 - recursion desired
- OP=0 - standard query
- QR=0 - this is a query


**Reply from Second NS to Resolver**

<table>
<thead>
<tr>
<th></th>
<th>TTL</th>
<th>protocol</th>
<th>header cksum</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>src IP</th>
<th>dst IP</th>
</tr>
</thead>
<tbody>
<tr>
<td>64.170.162.98</td>
<td>68.94.156.1</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>src port</th>
<th>dst port</th>
</tr>
</thead>
<tbody>
<tr>
<td>53</td>
<td>5798</td>
</tr>
</tbody>
</table>

QR=1 - this is a response
AA=1 - Authoritative!
RA=0 - recursion unavailable

**bailiwick checking:**
response is cached if it is within the same domain of query
(i.e. a.com cannot set NS for b.com)
Here we go again...

- What security checks are in place?
  - Random query ID’s to link responses to queries
  - Bailiwick checking (sanity check on response)
- No authentication
  - DNSsec is supposed to fix this but few DNS servers support it
- Many things trust hostname to IP mapping
  - Browser same-origin policy
  - URL address bar
Attacks against DNS?

- Corrupted nameservers
- Intercept & manipulate requests
- DDoS
- Cache poisoning
- Phishing / typo squatting / piggy-backing
DDoS against DNS

• Denial of Service
  – take down DNS server, clients can’t use Internet
  – Feb 6, 2007 attack against 6 of 13 root servers:
    • 2 suffered very badly
    • Others experienced heavy traffic

• DoD purportedly has interesting response:
  – “In the event of a massive cyberattack against the country that was perceived as originating from a foreign source, the United States would consider launching a counterattack or bombing the source of the cyberattack, Hall said. But he noted the preferred route would be warning the source to shut down the attack before a military response.”
  – http://www.computerworld.com/s/article/9010921/RSA_U.S._cyber_counterattack_Bomb_one_way_or_the_other
DNS cache poisoning

How might an attacker do this?
Assume DNS server uses predictable UDP port
2b. QID=1000
   IP for www.bankofsteve.com?

3. QID=1000
   referral to ns1.bankofsteve.com

4. QID=1001
   IP for www.bankofsteve.com?

5. QID=1001
   IP = 10.1.1.1

Simultaneously

- Flood of Forged answers:
  - IP = 10.9.9.99
  - QID=1000 - mismatch
  - QID=1001 - success!
  - QID=1002 - mismatch

6. www.bankofsteve.com is 10.9.9.99

1. IP for www.BankOfSteve.com?
Another idea:
- Poison cache for NS record instead
- Now can take over all of second level domain

How many tries does this require?
- Try 256 different QIDs
- Good chance of success
- Repeat if needed
Does happen in the wild

HD Moore pwned with his own DNS exploit, vulnerable AT&T DNS servers to blame

By Dancho Danchev | July 30, 2008, 8:08am PDT

**Summary:** A week after \(|)\)ruid and HD Moore release part 2 of DNS exploit, HD Moore’s company BreakingPoint has suffered a traffic redirection to a rogue Google site, thanks to the already poisoned cache at AT&T servers to which his company was forwarding DNS traffic: “It happened on Tuesday morning, when Moore’s company, BreakingPoint had some [...]”

Defenses

• Query ID size is fixed at 16 bits
• Repeat each query with fresh Query ID
  – Doubles the space
• Randomize UDP ports
• DNSsec
  – Cryptographically sign DNS responses, verify via chain of trust from roots on down
Phishing is a common problem

- **Typo squatting:**
  - www.qpple.com
  - www.goggol.com
  - www.nytimes.com

- **Other shenanigans:**
  - www.badguy.com/(256 characters of filler)/www.google.com

- **Phishing attacks**
  - These just trick users into thinking a malicious domain name is the real one
The page at goggle.com says:

Congratulations!

You are Today's Lucky Visitor.

Click OK to continue

---------------------------
WARNING!

YOUR COMPUTER MAY BE INFECTED:

System detected (2) Potentially Malicious Viruses. The data on your computer is NOT SAFE!

Your Personal & Financial Information IS NOT SAFE
To Remove Viruses, Call Tech Support Now:

855-521-0242

(24/7 - Toll free- High Priority Virus & Spyware Removal Call Line for Your IP Address: 128.105.35.160)

Clean Now!
An anonymous reader writes

"There is an interesting story over at the SANS Internet Storm Center that shows details on about 50 organizations that have had new machine names added to their DNS zone information. These were then pointed to sites used to boost the search engine cred of pharma, personals, and porn sites. If you outsource your DNS, how would you ever catch something like this?"
DNS piggybacking

<table>
<thead>
<tr>
<th>Piggy backed sites</th>
<th>IP</th>
<th>Main site</th>
<th>IP</th>
</tr>
</thead>
<tbody>
<tr>
<td>buy-cialis.sacmetrofire.ca.gov</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>buy-viagra.sacmetrofire.ca.gov</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>drugs.sacmetrofire.ca.gov</td>
<td>74.220.215.210</td>
<td><a href="http://www.sacmetrofire.ca.gov">www.sacmetrofire.ca.gov</a></td>
<td>66.147.240.176</td>
</tr>
<tr>
<td>mgdrugs.sacmetrofire.ca.gov</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>rxdrugs.sacmetrofire.ca.gov</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

From https://isc.sans.edu/diary/What+s+In+A+Name+/11770

Attackers maliciously added extra lower level domain names to valid domain name

This is helpful for search engine optimization
BGP and routing

wisc.edu

charter.net

BGP (exterior BGP)

defense.gov

OSPF within AS’s (Open shortest-path first)
BGP

• Policy-based routing
  – AS can set policy about how to route
    • economic, security, political considerations

• BGP routers use TCP connections to transmit routing information

• Iterative announcement of routes
• 2, 7, 3, 6 are Transit AS
• 8, 1 are Stub AS
• 4, 5 multihomed AS
• Algorithm seems to work OK in practice
  – BGP does not respond well to frequent node outages
IP hijacking

• BGP unauthenticated
  – Anyone can advertise any routes
  – False routes will be propagated

• This allows IP hijacking
  – AS announces it originates a prefix it shouldn’t
  – AS announces it has shorter path to a prefix
  – AS announces more specific prefix
Malicious or misconfigurations?

• AS 7007 incident in 1997
  – “Okay, so panic ensued, and we unplugged *everything* at 12:15PM almost to the second.” [sic]

• China Telecom hijacks large chunks of Internet in 2010
  – [http://bgpmon.net/blog/?p=282](http://bgpmon.net/blog/?p=282)
Youtube incident

• Pakistan attempts to block Youtube
  – youtube is 208.65.152.0/22
  – youtube.com = 208.65.153.238

• Pakistan ISP advertises 208.65.153.0/24
  – more specific, prefix hijacking

• Internet thinks youtube.com is in Pakistan

• Outage resolved in 2 hours...
BGPsec

- Route announcements must be cryptographically signed
  - AS can only advertise as itself
  - AS cannot advertise for IP prefixes it does not own
- Requires a public-key infrastructure (PKI)
- Still in development:
Internet Security

• Recurring themes:
  – Built without any authenticity mechanisms in mind
  – Functionality mechanisms (sequence #’s) become implicit security mechanisms
  – New attempts at (somewhat) backwards-compatible security mechanisms
    • IP -> IPsec
    • DNS -> DNSsec
    • BGP -> BGPsec