Penetration Testing in the Age of IPv6

Jayson Salazar
jsalazar@ernw.de

Rafael Schaefer
rschaefer@ernw.de
Road Map

- Why IPv6 Penetration Testing?
- Introduction to IPv6, Core Protocols
- Attack Surface of IPv6 Networks
- IPv6 Compared to IPv4
- Tools of the Trade
- DEMOS
- Conclusions
Why IPv6 Penetration Testing?

Increasingly popular and astonishingly complex
The IPv6 Vision

- Everything gets a networking interface!
- Personal appliances are increasingly incorporating networking capabilities.
- Concrete efforts are being directed towards materializing the “Internet of Things.”
- IPv6 deployment has been slowly but steadily taking off.
Two Questions come to Mind, though

- Is IPv6 **understood** sufficiently and **mature enough** for deployment?
- Do we have the **know-how** for securing such **shape-shifting networks**?

<table>
<thead>
<tr>
<th>Source</th>
<th>Destination</th>
<th>Protocol</th>
<th>Len</th>
<th>Info</th>
</tr>
</thead>
<tbody>
<tr>
<td>fe80::271::f06::54c2::6f0</td>
<td>ff02::1</td>
<td>ICMPv6</td>
<td>142</td>
<td>Router Advertisement from 80:71:1f:c2:05:f0</td>
</tr>
<tr>
<td>2001:67c:6ec::1620:f482:8175:abb1::6079</td>
<td>ff02::1:ffc2::6f0</td>
<td>ICMPv6</td>
<td>86</td>
<td>Neighbor Solicitation for fe80::271:ff06:54</td>
</tr>
<tr>
<td>fe80::1c0a::bba2:a026::0db</td>
<td>ff02::1:ffc2::6f0</td>
<td>ICMPv6</td>
<td>86</td>
<td>Neighbor Solicitation for fe80::271:ff06:54</td>
</tr>
<tr>
<td>2001:67c:6ec::1620:f482:8175:abb1::6079</td>
<td>ff02::1:ffc2::6f0</td>
<td>ICMPv6</td>
<td>86</td>
<td>Neighbor Solicitation for fe80::271:ff06:54</td>
</tr>
<tr>
<td>fe80::1c0a::bba2:a026::0db</td>
<td>ff02::1:ffc2::6f0</td>
<td>ICMPv6</td>
<td>86</td>
<td>Neighbor Solicitation for fe80::271:ff06:54</td>
</tr>
<tr>
<td>fe80::271::1f06::54c2::6f0</td>
<td>fe80::da9d:67ff:fe98:eca5</td>
<td>ICMPv6</td>
<td>86</td>
<td>Neighbor Solicitation for fe80::da9d:67ff:fe98:eca5</td>
</tr>
<tr>
<td>fe80::271::1f06::54c2::6f0</td>
<td>fe80::da9d:67ff:fe98:eca5</td>
<td>ICMPv6</td>
<td>78</td>
<td>Neighbor Advertisement fe80::271:ff06:54c2:6f0</td>
</tr>
<tr>
<td>2001:67c:6ec::1620:f482:8175:abb1::6079</td>
<td>ff02::1:ffc2::6f0</td>
<td>ICMPv6</td>
<td>86</td>
<td>Neighbor Solicitation for 2001:67c:6ec::1620:ff06:54</td>
</tr>
<tr>
<td>fe80::6154:4138::f5a::163</td>
<td>ff02::1:ffc2::6f0</td>
<td>ICMPv6</td>
<td>86</td>
<td>Neighbor Solicitation for fe80::271:ff06:54</td>
</tr>
<tr>
<td>2001:67c:6ec::1620:b476:56a5:142b::caac</td>
<td>ff02::1:ffc2::6f0</td>
<td>ICMPv6</td>
<td>86</td>
<td>Neighbor Solicitation for 2001:67c:6ec::1620:ff06:54</td>
</tr>
<tr>
<td>fe80::1c0a::bba2:a026::0db</td>
<td>ff02::1:ffc2::6f0</td>
<td>ICMPv6</td>
<td>86</td>
<td>Neighbor Solicitation for fe80::271:ff06:54</td>
</tr>
<tr>
<td>2001:67c:6ec::1620:da9d:67ff:fe98:eca6</td>
<td>ff02::1</td>
<td>ICMPv6</td>
<td>62</td>
<td>Echo (ping) request id=0x17c6, seq=0, hop l i</td>
</tr>
</tbody>
</table>
Introduction to IPv6

Protocols Running the Show
What’s New in IPv6? - I

- Several things have changed.
- Yes, the **HUGE address space** is the most well-known one.
- But, we also have the IPv6 **Extension Headers**
What’s New in IPv6? - II

- Router Advertisements and the **Neighbor-Discovery** protocol
- Multicasting plays a major role in IPv6
- There are new complex beasts such as the **Multicast Listener Discovery** protocol
Networking is still networking, **BUT**

- Bigger **address-space**, no **NAT** needed or possible
- **ICMP** was **overhauled**, is the basis for other protocols
- **Oversimplifying**, ND is to IPv6 what ARP was to IPv4
- **ND** encompasses other minor **sub-functionalities**
- ND is more complex than ARP
- MLD was created and plays a ‘major’ role in IPv6. It’s highly complex, often misunderstood and has some serious scalability issues.
- Half the action in IPv6 happens on the Local-Link
- So, what are the attack vectors in IPv6’s expanded attack surface?
A Look at the IPv4 and IPv6 Headers

Legend:
- Field's name kept from IPv4 to IPv6
- Field not kept in IPv6
- Name and position changed in IPv6
- New field in IPv6
ICMPv6 101

- First specified in **RFC 2462**, latest in **RFC 4443**.
- **ICMPv6** is an integral part of every **IPv6** implementation, the foundation of other protocols.

<table>
<thead>
<tr>
<th>Type(Value)</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Destination Unreachable (with codes 0,1,2,4)</td>
</tr>
<tr>
<td>2</td>
<td>Packet too big (Code 0)</td>
</tr>
<tr>
<td>3</td>
<td>Time Exceeded (Code 0,1)</td>
</tr>
<tr>
<td>4</td>
<td>Parameter Problem (Code 0,1,2)</td>
</tr>
<tr>
<td>128</td>
<td>Echo Request (Code 0)</td>
</tr>
<tr>
<td>129</td>
<td>Echo Reply (Code 0)</td>
</tr>
<tr>
<td>130</td>
<td>Multicast Listener Query</td>
</tr>
<tr>
<td>131</td>
<td>Multicast Listener Report</td>
</tr>
<tr>
<td>132</td>
<td>Multicast Listener Done</td>
</tr>
<tr>
<td>133</td>
<td>Router Solicitation</td>
</tr>
<tr>
<td>134</td>
<td>Router Advertisement</td>
</tr>
<tr>
<td>135</td>
<td>Neighbor Solicitation</td>
</tr>
<tr>
<td>136</td>
<td>Neighbor Advertisement</td>
</tr>
<tr>
<td>137</td>
<td>Redirect</td>
</tr>
</tbody>
</table>
Neighbor Discovery 101

- IS the soul of the Local-Link
- ND’s duties:
  - Neighbor Discovery
  - Router Discovery
  - Prefix Discovery
  - Parameter Discovery
  - Address auto-configuration
  - Next-Hop Determination
  - Duplicate Address Detection
The **Querier** sends *periodical Queries* to which Listeners with reportable addresses reply.

- The **Querier** does not learn which or how **many** clients are interested in which sources.

- The **Querier** uses reported information for deciding what **ingress data** to **forward**.

Anyone expecting this data?

Me, let it through!
Attack Surface in IPv6 Networks

IPv6, a Fancy Code-Word for Excruciating Complexity
Host-Level Discrepancies

- Unexpected **differences** in kernels and IPv6-Stacks behavior.
  - Should packets with source-address 1 be processed on an external interface?
- These differences **lead** to **lack** of awareness with respect to **IPv6 hardening in different platforms**.
- Also, **services** must often be **configured differently**. Hence, **admins** usually **slip**. E.g. services listening on all IPv6 capable interfaces.
Even Applications Behave Differently

- **Applications** working appropriately in IPv4 usually lack **IPv6 security capabilities**, mostly due to having been untested.

- One such example is the **Filezilla** server, whose **autoban** functionality **doesn’t work with IPv6**.

- [http://blog.webernetz.net/2014/05/14/filezilla-server-bug-autoban-does-not-work-with-ipv6/](http://blog.webernetz.net/2014/05/14/filezilla-server-bug-autoban-does-not-work-with-ipv6/)
All Black-Listing approaches to security controls have a hard time in IPv6 networks.

Mostly due to extension-headers and fragmentation.

But also because of ambiguities in the RFCs.

This makes possible the evasions of IDPS devices and security mechanisms such as DHCPv6 Guard and RA-Guard.
Don’t Forget Profiting from the Protocols

- ICMPv6, ND and MLD are perfect candidates for performing reconnaissance.
- Complex protocols with complex packet structures such as MLD make perfect targets for performing DoS attacks.
- A poorly hardened Local-Link in an IPv6 network makes leveraging ND for malicious purposes, e.g. MitM attacks.
By-Passing ACLs

- ACLs are most effective when the characteristics of undesired behavior are clear.
- IPv6 provides a great deal of flexibility, one does not have to be content with a ‘standard deployment’.
- However, this very flexibility is one major enemy of ACLs based filtering.
- Which packets should be rejected?
  - Those coming from a certain address?
  - With one extension-header or two?
  - Fragmented or not fragmented?
Fiddling with ND Messages

- Fill, and keep filled, the Neighbor-Cache of a legitimate host in the network.
- Reply with spoofed Neighbor-Advertisements to Neighbor-Solicitations.
- Unsolicited Spoofed Neighbor-Advertisements and Neighbor-Solicitations.
- Flooding hosts and causing a DoS consumption due to poorly implemented IPv6 stacks.
- Remember, the Local-Link is “trustworthy”
Playing with Router Advertisements

- **Router-Advertisements** are, as part of auto-configuration approach, **fundamental** part of IPv6.
- Once again, the **Local-Link** is considered **trustworthy**!
- A potential **attacker** can send **Rogue-RAs** into the network in order to cause **DoS** conditions or redirect traffic due to **host** using the **information** contained therein.
- **Lots of DoS conditions** to be found here!
IPv6 Compared to IPv4
The Good, the Bad and the Ugly
### IPv6 vs. IPv4 some Numbers

<table>
<thead>
<tr>
<th>Alexa Top</th>
<th>IPv6 enabled</th>
<th>Prozent</th>
</tr>
</thead>
<tbody>
<tr>
<td>10</td>
<td>5</td>
<td>50%</td>
</tr>
<tr>
<td>100</td>
<td>32</td>
<td>32%</td>
</tr>
<tr>
<td>1000</td>
<td>162</td>
<td>16.2%</td>
</tr>
<tr>
<td>10000</td>
<td>955</td>
<td>9.55%</td>
</tr>
<tr>
<td>100000</td>
<td>8030</td>
<td>8.03%</td>
</tr>
<tr>
<td>500000</td>
<td>35041</td>
<td>7.01%</td>
</tr>
<tr>
<td>1.000.000</td>
<td>...</td>
<td>...</td>
</tr>
</tbody>
</table>
So, What do we Find when we Look Further?

<table>
<thead>
<tr>
<th>IPv6</th>
<th>PORT</th>
<th>STATE</th>
<th>SERVICE</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>22/tcp</td>
<td>open</td>
<td>ssh</td>
</tr>
<tr>
<td></td>
<td>80/tcp</td>
<td>open</td>
<td>unknown</td>
</tr>
<tr>
<td></td>
<td>111/tcp</td>
<td>filtered</td>
<td>rpcbind</td>
</tr>
<tr>
<td></td>
<td>443/tcp</td>
<td>open</td>
<td>unknown</td>
</tr>
<tr>
<td></td>
<td>2381/tcp</td>
<td>filtered</td>
<td>unknown</td>
</tr>
<tr>
<td></td>
<td>5666/tcp</td>
<td>filtered</td>
<td>unknown</td>
</tr>
<tr>
<td></td>
<td>8301/tcp</td>
<td>open</td>
<td>unknown</td>
</tr>
<tr>
<td></td>
<td>9070/tcp</td>
<td>open</td>
<td>unknown</td>
</tr>
<tr>
<td></td>
<td>9080/tcp</td>
<td>filtered</td>
<td>unknown</td>
</tr>
<tr>
<td></td>
<td>9090/tcp</td>
<td>filtered</td>
<td>zeus-admin</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>IPv4</th>
<th>PORT</th>
<th>STATE</th>
<th>SERVICE</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>80/tcp</td>
<td>open</td>
<td>http</td>
</tr>
<tr>
<td></td>
<td>443/tcp</td>
<td>open</td>
<td>https</td>
</tr>
</tbody>
</table>
Attacking Node Provisioning

- IPv4 has been more or less a stable for the last decade.
- This isn’t the case with IPv6
- IPv6’s vision is one of automation, where your fridge can easily join the cyber-party called IoT.
- But, what happens when said devices present heterogeneous behavior?
- What always happens ... the network breaks!
There are several IPv6 Stacks

<table>
<thead>
<tr>
<th>Category</th>
<th>RFC 1970</th>
<th>RFC 2410</th>
<th>RFC 4861</th>
<th>RFC 6980</th>
</tr>
</thead>
<tbody>
<tr>
<td>Neighbor Discovery</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Address Selection</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Generation of IID</td>
<td>EUI-64</td>
<td>Privacy Extensions</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Others</td>
<td></td>
<td></td>
<td></td>
<td>RFC 7217</td>
</tr>
</tbody>
</table>
What can we do about It?

Nothing!
What can we do about It?

- Read the specifications of your core devices!
- Ask the vendors for their REAL security features
- Harden your network

- IPv6 IS NOT plug-and-play!
- Stay updated with regard to IPv6:
  ERNW’s hardening guides for IPv6
Why is IPv6 so Hard?

- Trust model and **automatized** provisioning.
- **Complexity**
- **Lack of awareness and understanding** of the technologies involved
- Stack **heterogeneity**
- **Limited resources** available to **defenders**
Tools of the Trade

How to Interact with the IPv6 Stack
Profiting from IPv6 for Reconnaissance

- Leverage ICMP as usual, ICMPv6.
- IPv6 has ‘done away with broadcasting’, employ multicasting for host discovery.
- There’s one protocol we haven’t talked about, MLD. Every IPv6 host must reply to and process messages associated with the Multicast-Listener-Discovery protocol.
- Fragmentation can help with tricking systems into replying to ICMPv6 ECHO-Requests.
Some Well-Known Attacking Frameworks

- The Hackers’ Choice THC-IPv6 framework
  - https://www.thc.org/thc-ipv6/
- Si6 Networks IPv6-Toolkit
- Anonios Atlasis’ Chiron
  - http://www.secfu.net
- Although they somewhat overlap, they also complement each other.
The Hackers’ Choice IPv6 Toolkit

- A rich set of tools allowing certain interactions with IPv6 and its associated protocols.
- Although easy to use, it can hardly be customized
- Some interesting tools:
  - alive6
  - dnsrevnum6
  - ndpexhaust
  - fake_router
  - flood_router
  - fake_advertise6
The Chiron IPv6 Testing Framework

- **Chiron** offers several modules geared towards different potential attack vectors:
  - IPv6 Scanner
  - IPv6 Link-Local Message Creator
  - IPv4-to-IPv6 Proxy

- Makes no decisions for you regarding the validity of the packets, it simply is IPv6-aware.

- Really flexible, but due to being written in Python and based on Scapy can be easily customized.
IPv6 host fingerprinting is a bit immature but does the job most of the time.

Useful plugins:
- Targets-ipv6-multicast-mld
- IPv6-ra-flood
- Targets-ipv6-multicast-invalid-dst
- Targets-ipv6-multicast-echo
- IPv6-node-info
- Resolveall
More like, **Internet of Broken Things**!
- If they are **connected** they have an **IPv6 stack**
- If they have an **IPv6 stack** they have **data buffers**
- If they have **data buffers**, **someone slipped up**
- If **someone slips**, attackers **profit**

**Fuzzing** IPv6 stacks is incredibly important for empirically assessing the robustness of devices we rely on.

http://core0.staticworld.net/
Several reconnaissance and post-exploitation modules support IPv6

- It isn’t any harder than in IPv4
- Useful IPv6 modules:
  - auxiliary/gather/dns_srv_enum
  - auxiliary/scanner/discovery/ipv6_multicast_ping
  - auxiliary/scanner/discovery/ipv6_neighbor
  - auxiliary/scanner/discovery/ipv6_neighbor_router_advertisement
  - Good number of IPV6 payload-handlers for Meterpreter
Enough networking, what do we do web-penetration testing with?

- There are several alternatives:
  - As usual, BURP
  - Arachni for automated tests
  - SQLMap for your post-exploitation needs
  - For getting the big picture, Nessus

For more information see: Penetration Testing Tools that Support IPv6
DEMO I – Behind the Iron Curtains
Evading IDPS Devices with Fragmentation
IDPSS Evasion – The Scenario

Traffic watched by IDPS

Gateway

IDPS

Internet
DEMO II – No Video-Conferencing for You

Abusing MLD to trigger DoS conditions in Routers
MLD - The Scenario

Sender A
2001:DB8:2::C001

Receiver n
2001:DB8:1::C001

Attacker

PIM Traffic ——> MLD Traffic ——>
As anything in InfoSec, Stay Informed

- Our Blog with #IPv6 filter, insinuator.net
- IPv6 hackers’ mailing-list
- IETF mailing-list, our personal favorite v6Ops.
- NANOG’s mailing-list
Conclusions

- **Developments** are still taking place within the IPv6 specification; to deal with IPv6 is to deal with change and the associated security risks.
- **Complexity Kills!**
- IPv6 is not IPv4 with a longer address space, they differ greatly.
- Since understanding is the father of situational awareness, and situational awareness is the mother of security, study and understand IPv6!
Some Resources for those Interested in More

- Regarding tools, this ERNW Newsletter is a good start: Penetration Testing Tools that Support IPv6
- For guidance with respect to hardening IPv6 networks, NIST’s Guidelines for the Secure Deployment of IPv6
- TNO’s Testing the Security of IPv6 Implementations offers a good, albeit in some cases exaggerated, overview of attack vectors present in IPv6.
- For thorough study of IPv6 security and its intricacies, Hagen’s, Cisco’s or Microsoft’s books should do.
Thanks for your Time!

Enjoy Amsterdam!

Jayson Salazar
jsalazar@ernw.de

Rafael Schaefer
rschaefcr@ernw.de