Security Visualization Tools and IPv6 Addresses

David Barrera
P.C. van Oorschot

Carleton UNIVERSITY
Canada's Capital University

Introduction

- IPv4 address exhaustion predicted to happen in mid-2012 (745 days remaining) [1]
- 10% of Class A's remaining
- IPv6 deployment on the global Internet is low (~1%)
- IPv6 deployment inside organizations could be significantly higher

[1] http://www.potaroo.net/tools/ipv4/index.html

Introduction

- IPv6 is already here
- Userspace applications require almost no modifications
- Enabled in new operating systems
- Widely deployed in network tools

Introduction

- Visualization of IPv6 addresses is difficult
  - Really long numbers (128-bit)
  - Address space is sparsely populated
  - Transition mechanisms interfere with ‘real’ IPv6 addresses
- Most visualization tools don’t support IPv6
  - Assume IPv4
  - Tools are hardcoded to 32-bits
  - Drop/ignore IPv6 packets
  - Crash

Address Representation

- 2001:0DB8:0000:0078:9ABC:0000:0000:0000
- 2001:0DB8:0:0078:9ABC:0:0:0
- 2001:DB8:0:78:9ABC::

Packet Header Changes

- The IPv6 header omits rarely used fields
- IPID, flags, fragment offset, header checksum no longer present
- Flow label is new to the IP header

Packet Header Changes

	0 - 3	4 - 7	8 - 15	16 - 18	19 - 31
0	Version	Header Length	Type of Service (ToS)	Total Length	
32	Identification (ID)			Flags	Fragment Offset
64	Time to Live TTL		Protocol	Header Checksum	
96	Source IP Address				
128	Destination IP Address				
160					
192					
224					
256					**IPv4**

	0 - 3	4 - 11	12 - 15	16 - 23	24 - 31
0	Version	Traffic Class	Flow Label		
32	Payload Length			Next Header	Hop Limit
64					
96	Source IP Address				
128					
160					
192					
224	Destination IP Address				
256					
288					**IPv6**

Updated protocols

- DHCPv6 and address auto-configuration
- ICMPv6
- Multicast, scope IDs
- New security issues to visualize

Visualization Steps

- Capture raw data
- Parse
- Process and reformat
- Display

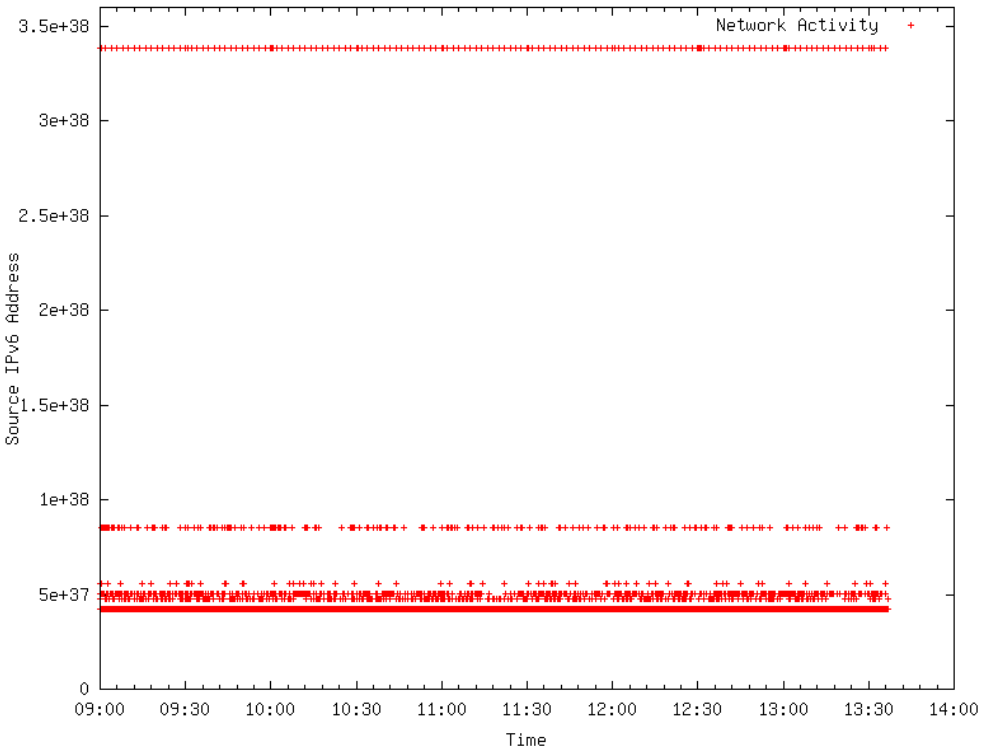
Visualization Steps

- Capture raw data - Wireshark, TCPdump
- Parse - Scripts in perl, python
- **Process and reformat** - Nonexisting fields, size of data structures
- **Display** - Designed for 32-bits

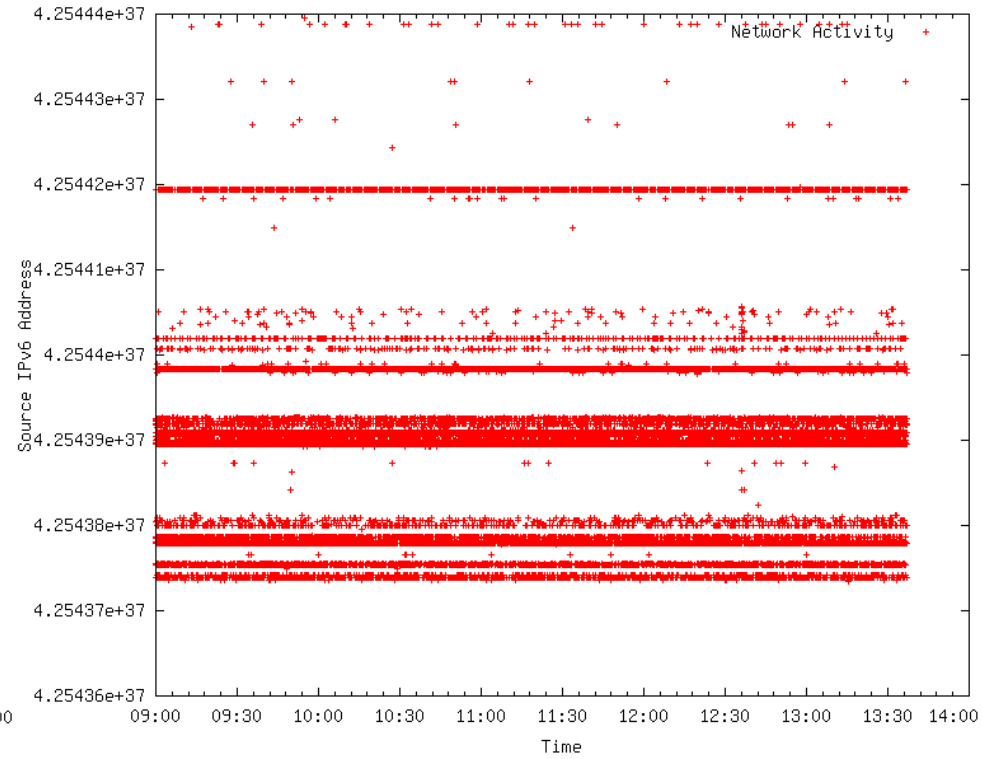
Whitespace Filtering

- The IPv6 address space is sparsely populated
- Vast majority of the address space is whitespace (darkspace)

Whitespace Filtering



(a) Normal view

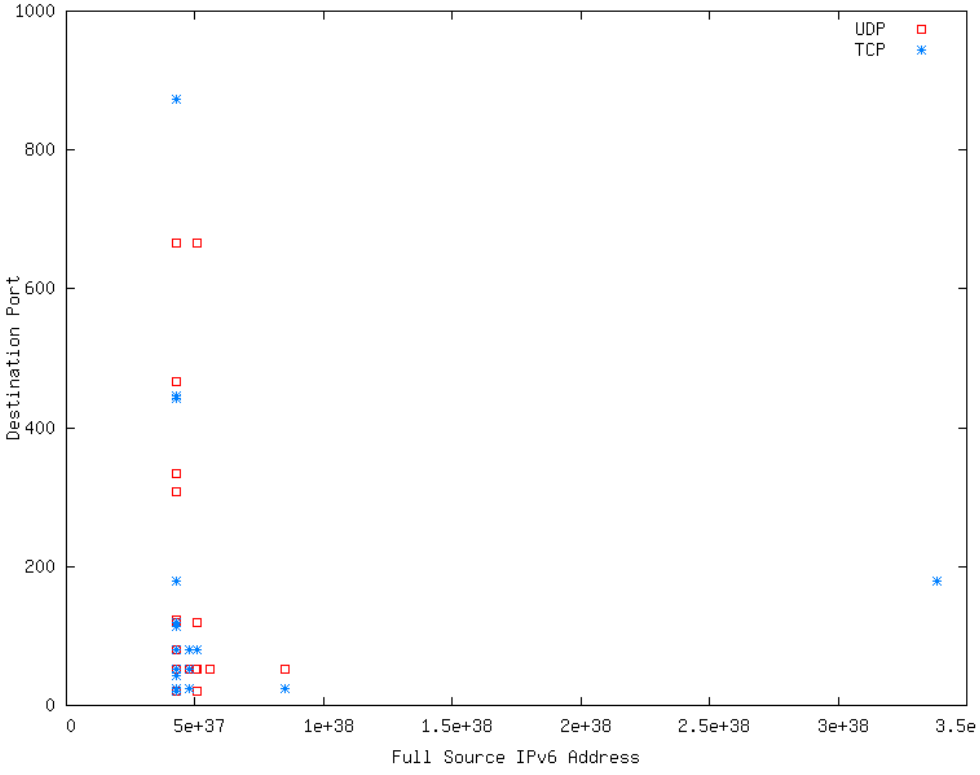


(b) 70000x zoom

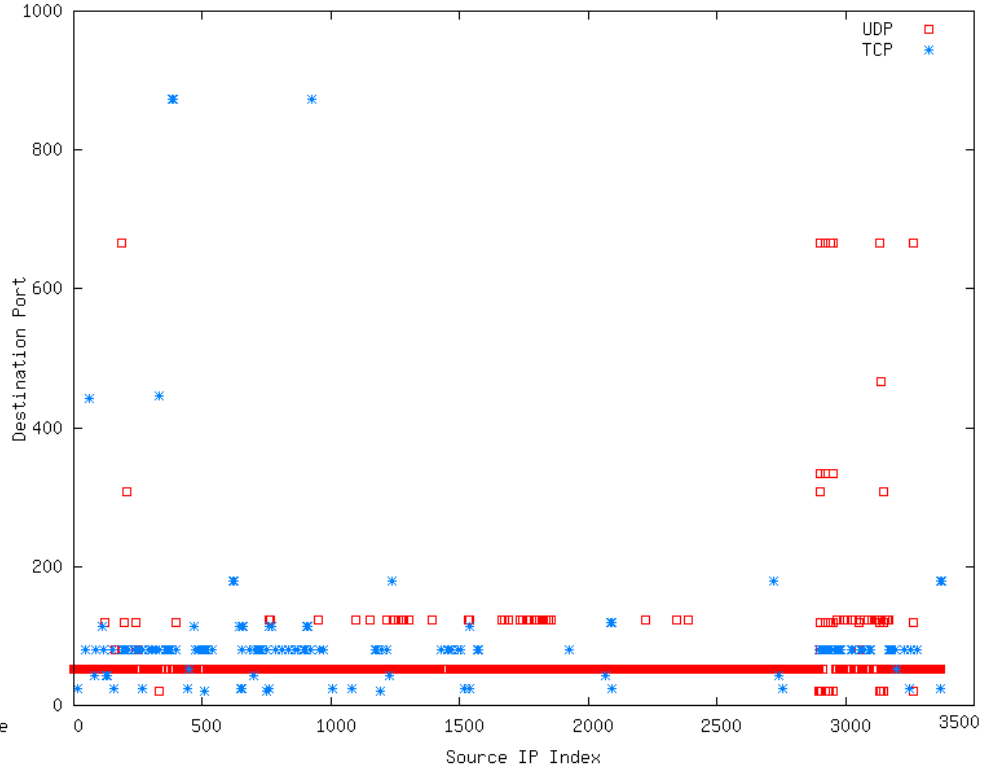
Whitespace Filtering

- Why is visualizing the entire address space important?
- Remove whitespace
  - Keep a (sorted) list of the “seen” IPv6 addresses
  - When plotting, use the index rather than the full address
  - Optionally insert gaps between points

Whitespace Filtering



(a) Original



(b) Whitespace Removed

IPv6 Address Allocation

- IPv6 addresses must be allocated hierarchically
  - Avoid unnecessary load on backbone routers
  - Give greater meaning to IP addresses
  - Flexibility to organizations for block assignments
- RFCs that specify how addresses can be aggregated to keep routing tables efficient
- Reading more bits of an IPv6 address reveals more information (country, AS, ISP, zone, etc)
- Treemaps are useful for visualizing hierarchy

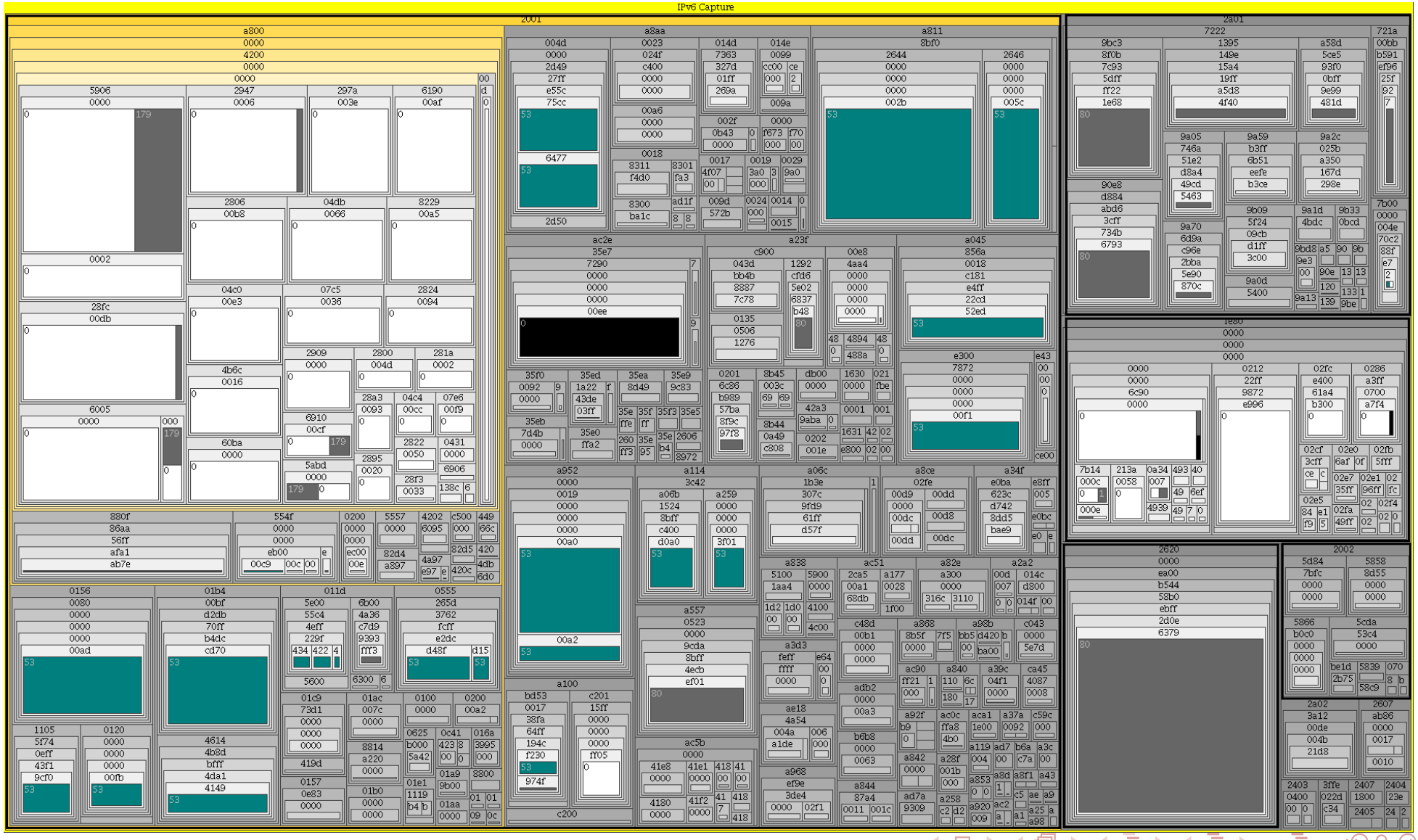
IPv6 Address Visualization with Treemaps

- Parse the dataset identifying all the unique IPv6 addresses
- Make each hextet a level of the tree
  - **2001:0DB8**:0000:0078:9ABC:0000:0000:0000
  - **2001:0DB8**:FABC:0078:9ABC:1234:5678:EEFF
- If there is more screen real-estate, display more hextets

IPv6 Address Visualization with Treemaps

- Improvements by color-coding type of traffic
- Make the size of each node proportional to the volume of data
- Display port number information in the contents of each node

IPv6 Treemaps



IPv6 Treemaps

Conclusions

- IPv6 is here
- New visualization techniques/tools to support IPv6
- 2 proposals for dealing with these datasets

Questions