The Matter of Heartbleed

Zakir Durumeric, Frank Li, James Kasten, Johanna Amann,
Jethro Beekman, Mathias Payer, Nicolas Weaver, David Adrian,
Vern Paxson, Michael Bailey, J. Alex Halderman

University of Michigan, University of California—Berkeley
University of Illinois—Urbana Champaign
International Computer Science Institute
Purdue University

Heartbleed Vulnerability

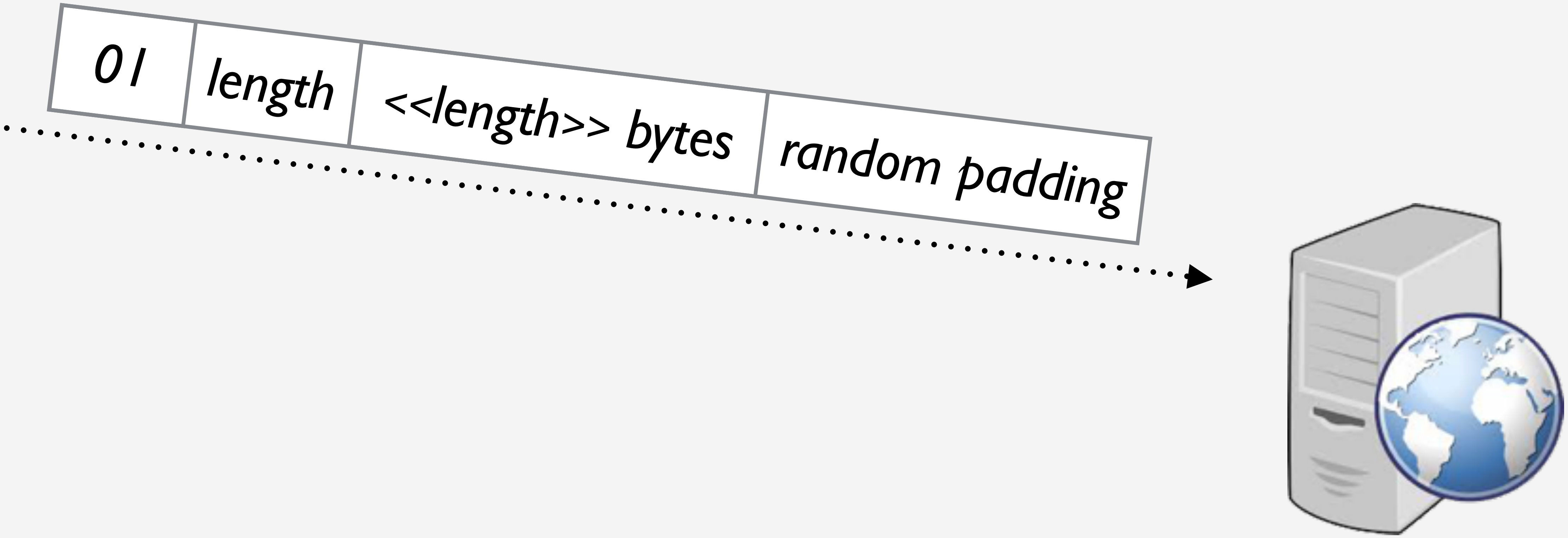
In April 2014, OpenSSL disclosed a catastrophic bug in their implementation of the TLS Heartbeat Extension

Vulnerability allowed attackers to dump private cryptographic keys, logins, and other private user data

Potentially effected any service that used OpenSSL for TLS—including web, mail, messaging and database servers

An estimated 24-55% of HTTPS websites were initially vulnerable

TLS Heartbeat Extension



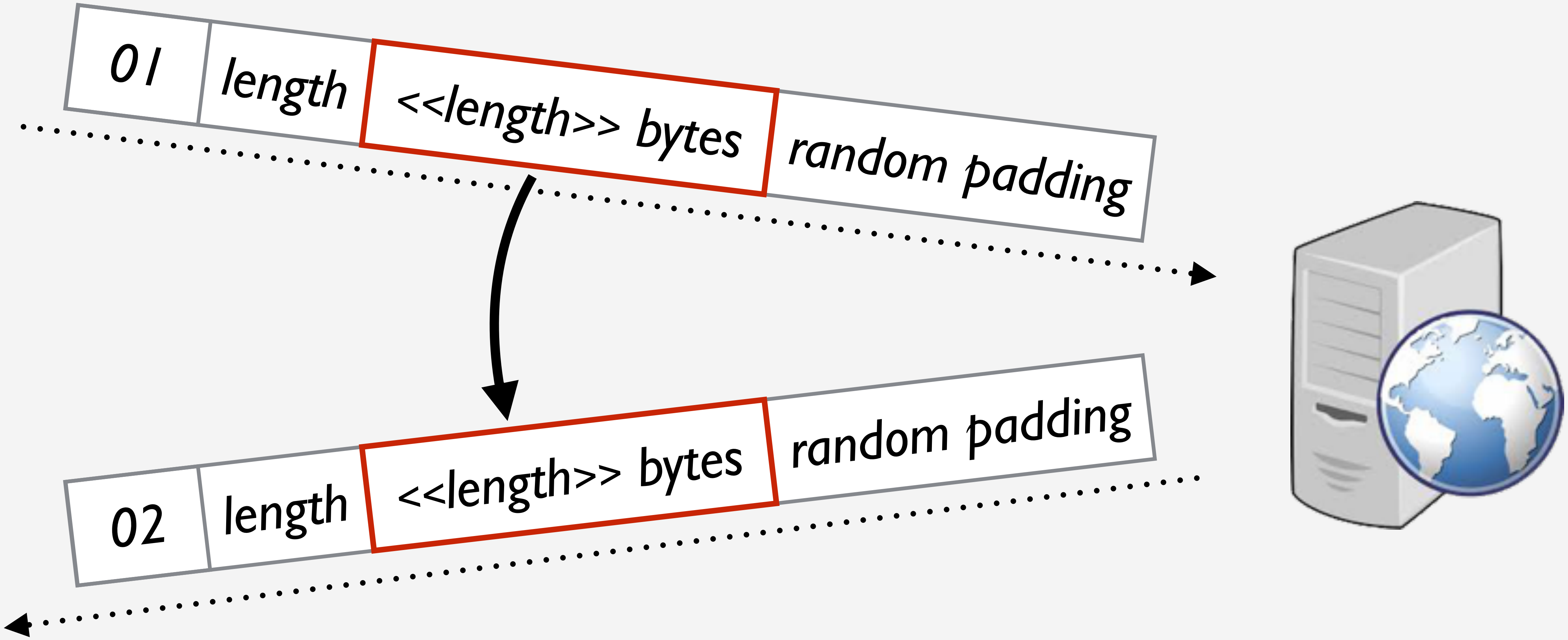
01

length

<<length>> bytes

random padding

TLS Heartbeat Extension



01

length

<<length>> bytes

random padding

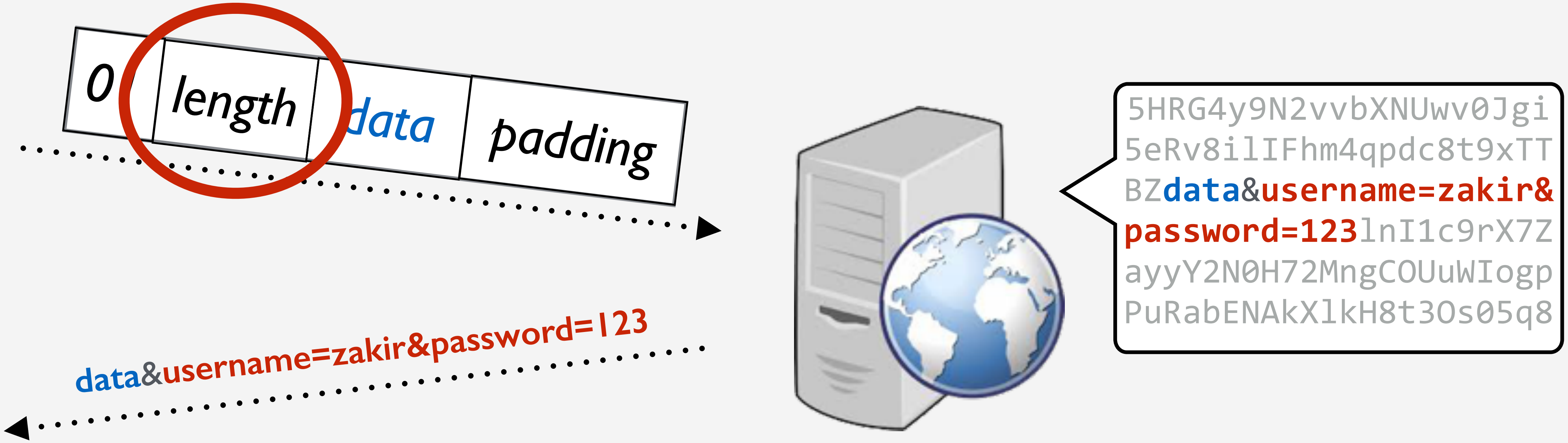
02

length

<<length>> bytes

random padding

Heartbleed Vulnerability



```
5HRG4y9N2vvbXNUwv0Jgi
5eRv8ilIFhm4qpdc8t9xTT
BZdata&username=zakir&
password=123lnI1c9rX7Z
ayyY2N0H72MngCOUuWIogp
PuRabENAkXlkH8t3Os05q8
```

Heartbleed Vulnerability: server trusts user provided length field and echoes back memory contents following request data

OPENVPN™
MAIL
NGINX
DOVECOT
Apache
Wi Fi™
Tor
PostgreSQL
jetty://
mongoDB
zimbra®
MySQL™
ProFtpd
Jabber.org
openswan™
sendmail.org
OpenLDAP
LIGHTTPD
fly light.
bitcoin

Database Servers

Messaging Servers

Crypto Currencies

Web Servers

POP3/IMAP Servers

SMTP Servers

Tracking the Vulnerability

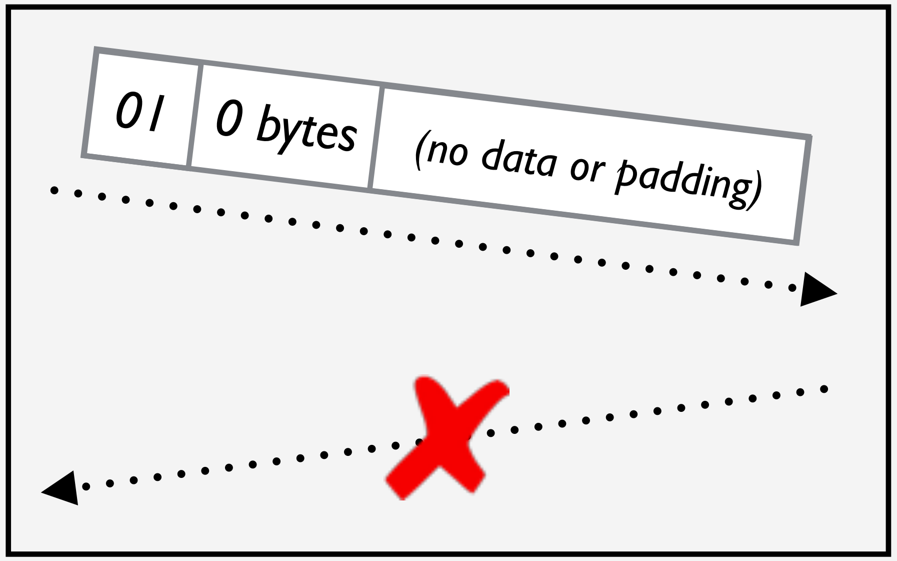
Data Collection

- Began scanning 48 hours after public disclosure
- Scanned Alexa Top 1 Million and 1% samples of IPv4 every 8 hours

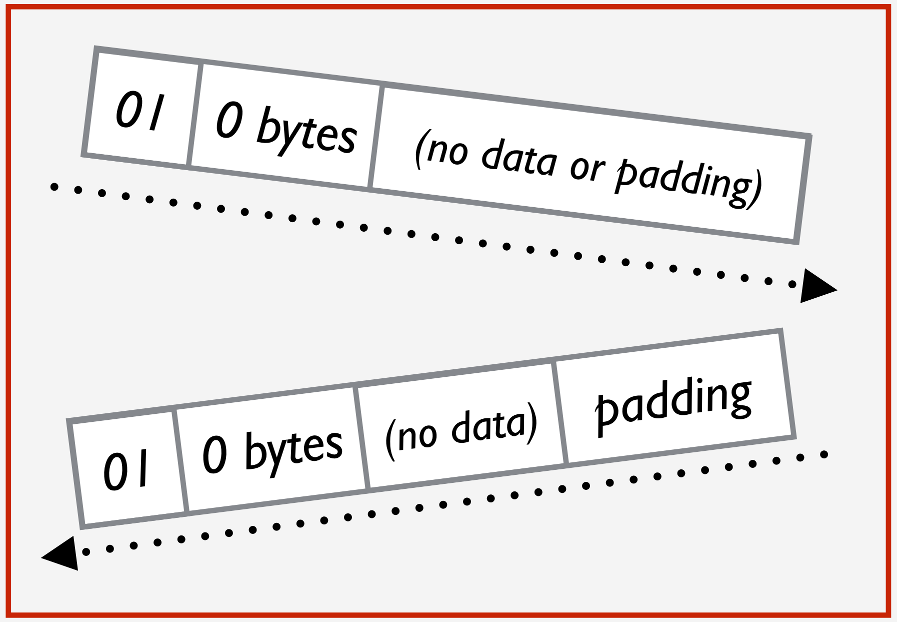
Scanning for Heartbleed

- Modified ZMap to scan for vulnerable versions of OpenSSL
- Instead of exploiting the vulnerability, we checked for non-compliant behavior of vulnerable OpenSSL version

RFC 6520 Compliant



Vulnerable OpenSSL



Tracking the Vulnerability

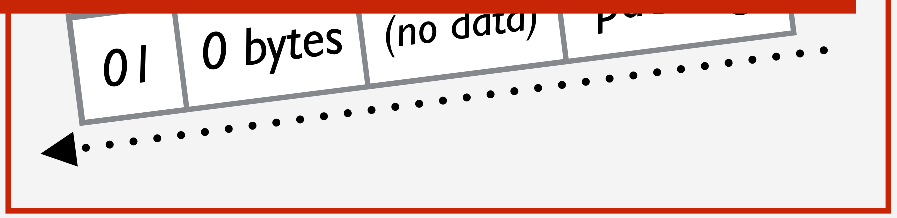
RFC 6520 Compliant

Data Collection

S

We did not exploit Heartbleed Vulnerability—no private memory is ever sent back by the server

of vulnerable OpenSSL version



Top 100 Websites

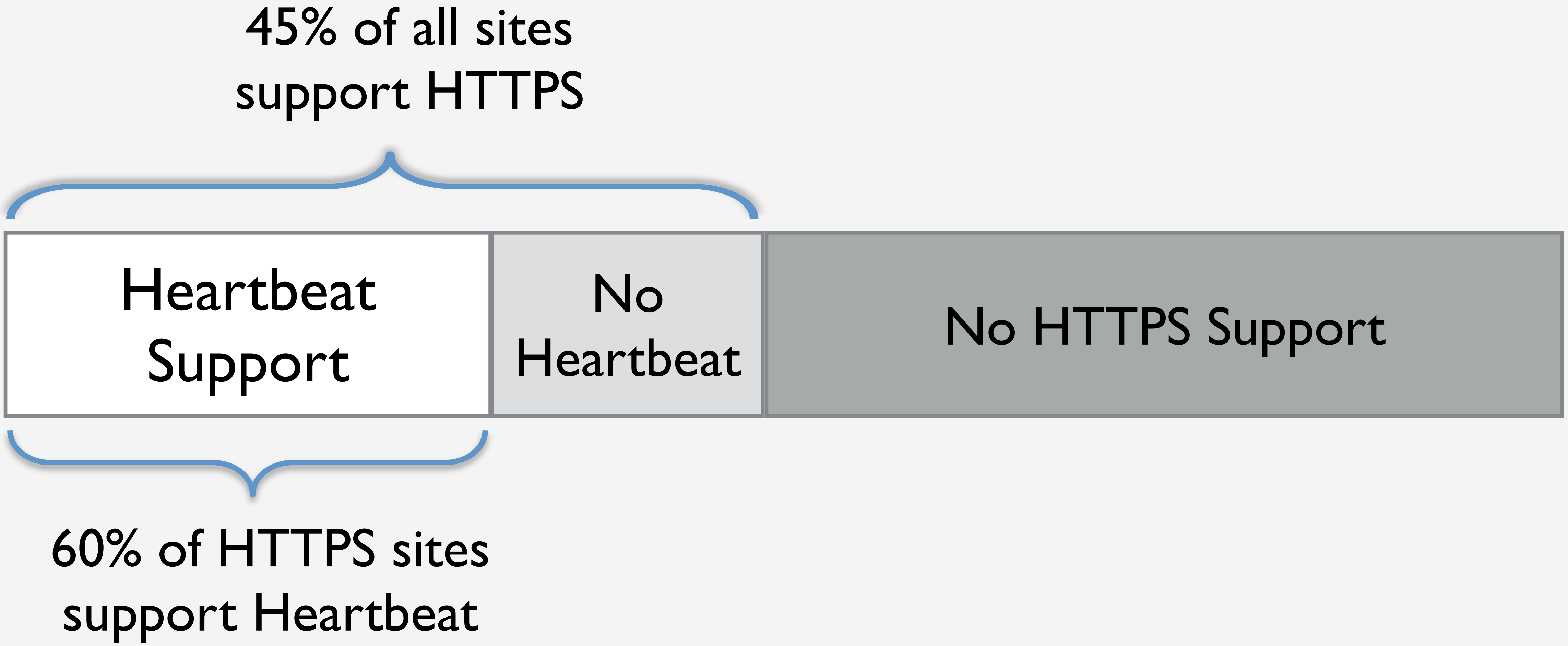
By aggregating press releases and others' scans, we found evidence that at least 44 of the Top 100 sites were initially vulnerable

A small handful of sites remained vulnerable at 24 hours—including Yahoo, Imgur, Stack Overflow, Flickr, Sogou, Ok Cupid, and Duck Duck Go

The Top 500 sites were patched within 48 hours—when we began our regular scans

Our First Scan — Disclosure + 2 Days

Unclear who was initially vulnerable beyond the Top 100—little attention was paid to the extension prior to public disclosure



45% of all sites support HTTPS

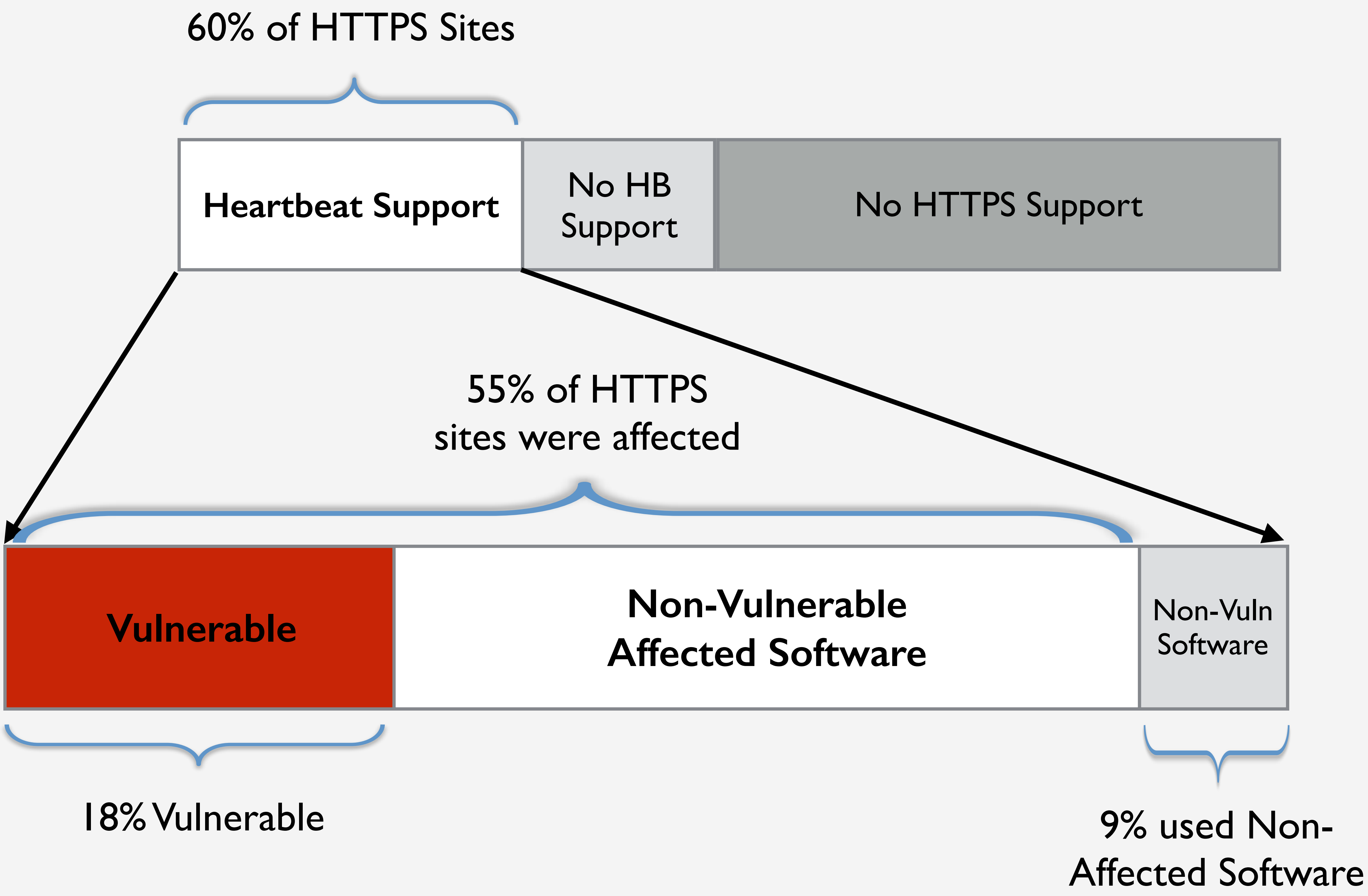
Heartbeat Support

No Heartbeat

No HTTPS Support

60% of HTTPS sites support Heartbeat

Our First Scan — Disclosure + 2 Days



Estimating Initial Impact

No Scans in the first 48 hours — how do we estimate initial impact?

Upper Bound

If all the servers that support Heartbeat and used affected software were initially vulnerable—55% of HTTPS sites were affected

Lower Bound

TLS 1.1 and 1.2 were introduced along with Heartbeat in OpenSSL 1.0.1

32.6% of sites supported TLS 1.1 or 1.2, of which 73% use affected software

We estimate that 24-55% of Alexa sites were initially vulnerable

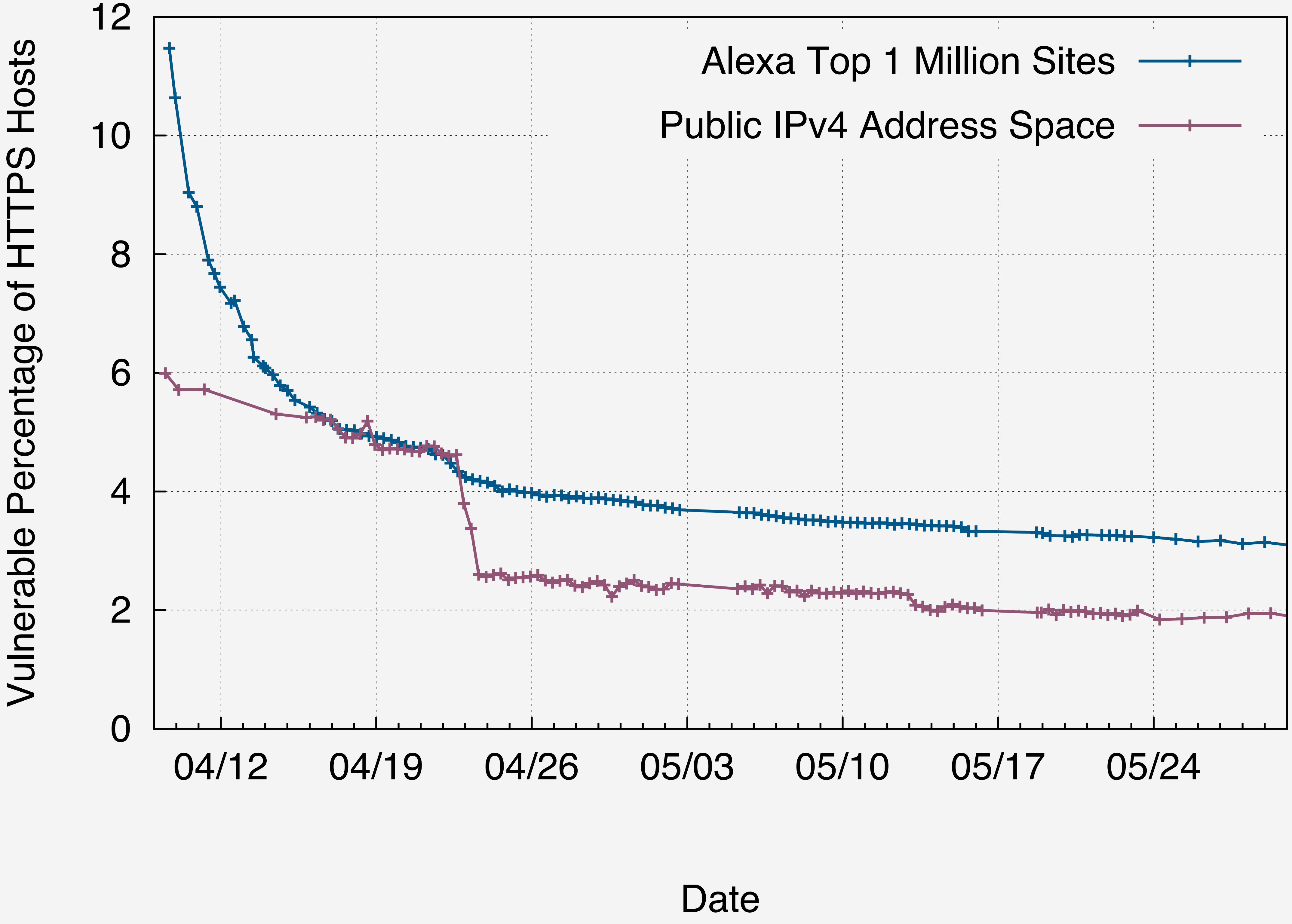
What about the rest of the Internet?

11% of IPv4 HTTPS hosts supported Heartbeat

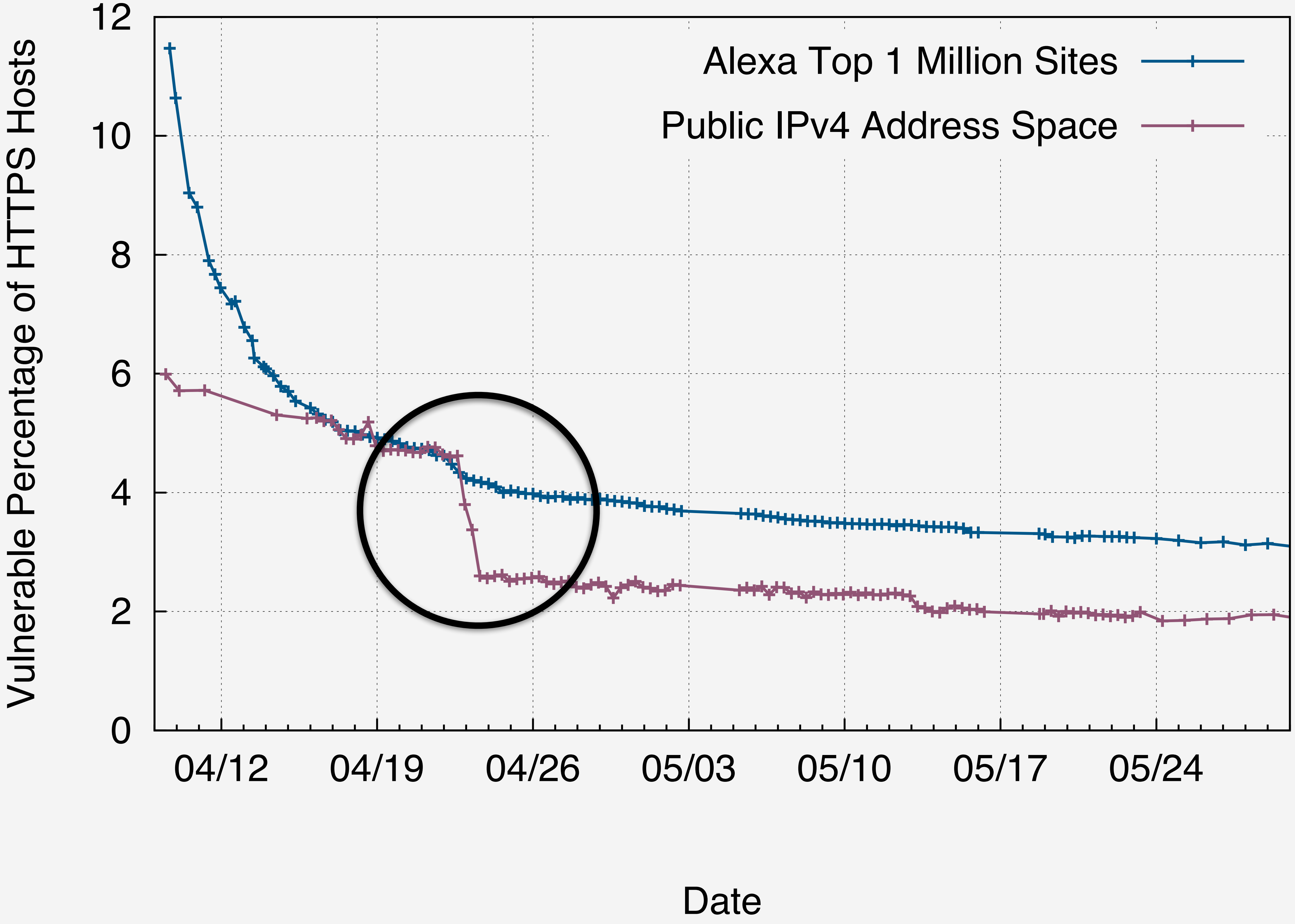
6% of HTTPS hosts were vulnerable

We investigated clusters of similar certificates and found 74 common vulnerable devices

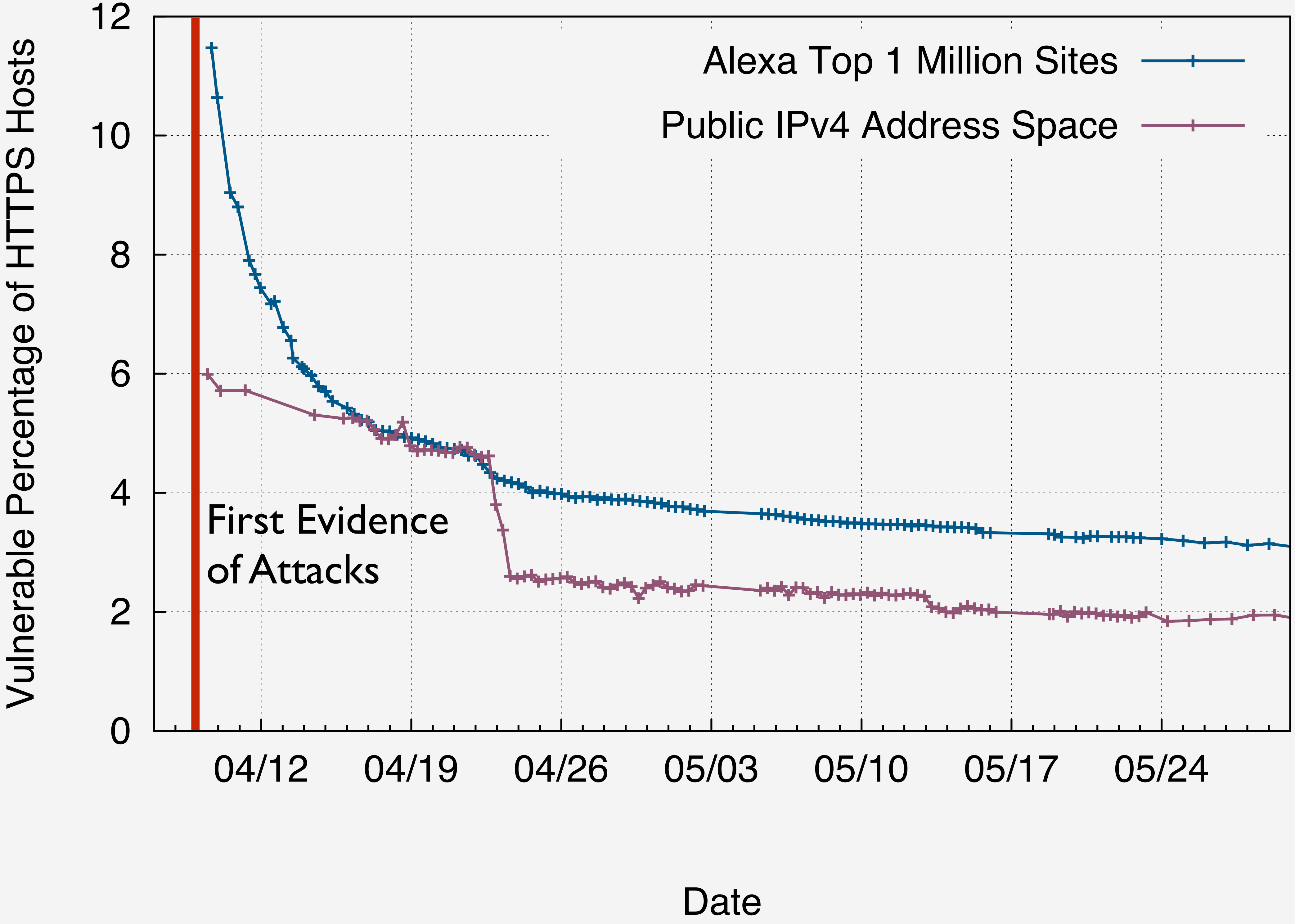
Patching Behavior



Patching Behavior



How fast is fast enough?



Attack Scene

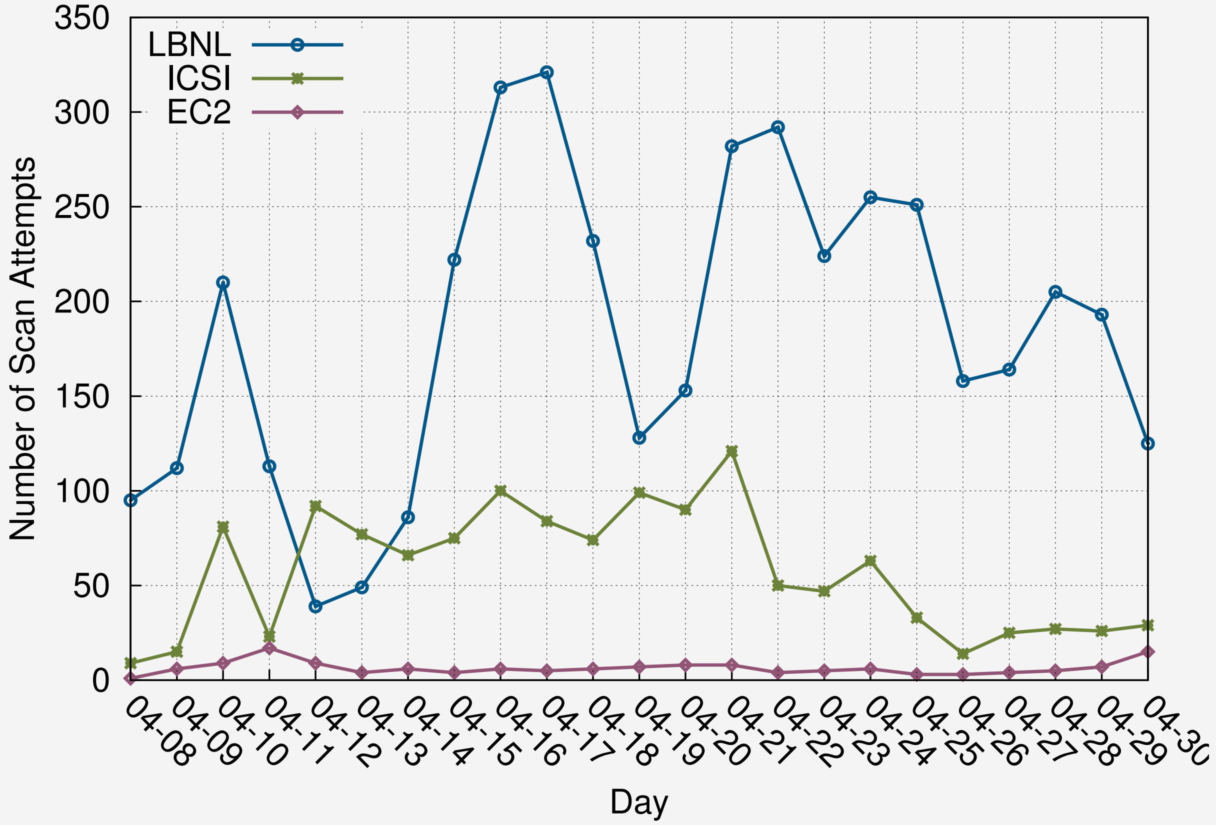
We examined packet traces from Lawrence Berkeley National Laboratory (LBNL), the International Computer Science Institute (ICSI), and an Amazon EC2 honeypot

No evidence of attack prior to disclosure

We detected the first scan traffic 22 hours after disclosure from University of Latvia

In total, we observed 6,000 probe attempts from 692 hosts

Two major outliers—*filippio.io* (3,964 attempts from 40 hosts) and *ssllabs.com* (16 attempts from 5 hosts)



Attack Scene

Only 11 hosts scanned both EC2 honeypot and ICSI network

Only 6 hosts scanned more than 100 hosts at ICSI—Michigan, TU Berlin, Chinanet (2), Nagravision, and Rackspace

Appears to be little Internet-Wide scanning.

201 hosts scanned EC2 honeypot—attackers are likely targeting cloud providers' address space

AS Name	Scans
Amazon.com	4,267
China Telecom	507
China169 Backbone	147
Chinanet	115
University of Michigan	92
SoftLayer	85
University of Latvia	50
Rackspace	47
GoDaddy.com	34

Global Vulnerability Notifications

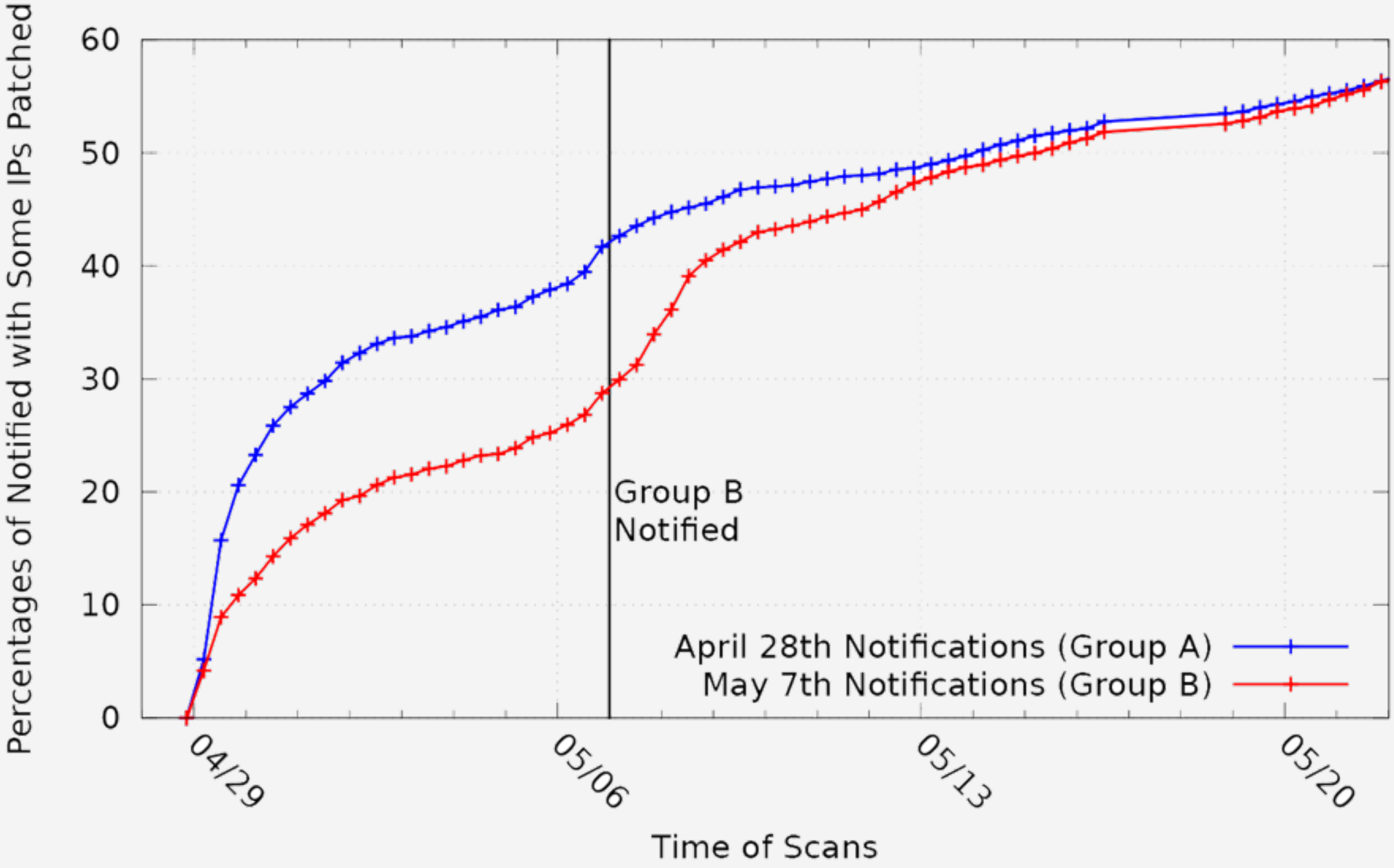
Two weeks post disclosure, nearly 600,000 hosts remained vulnerable

We contacted network administrators for non-embedded devices

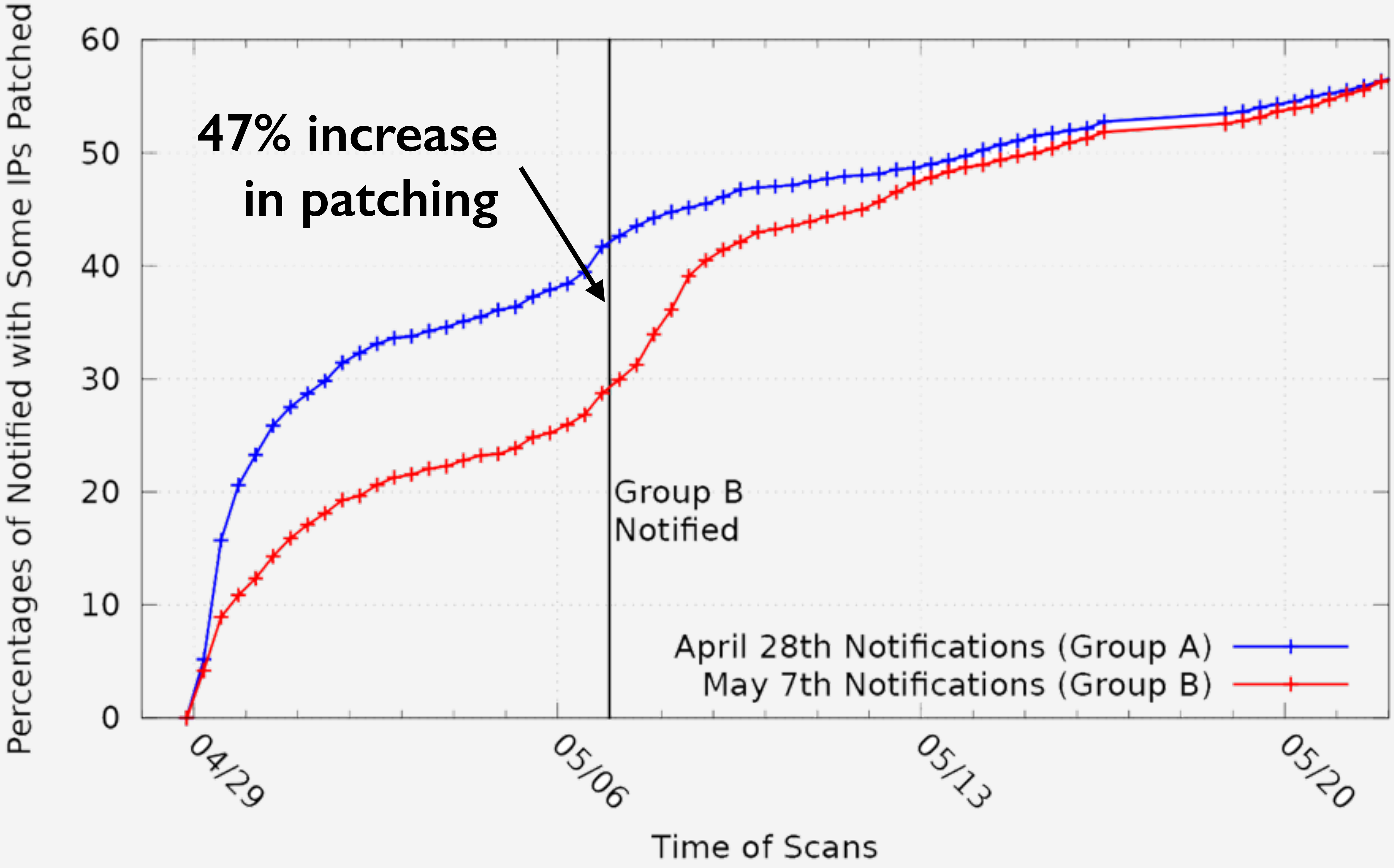
We aggregated vulnerable hosts by WHOIS abuse contact (4,648 distinct contacts)

Split abuse contacts into two groups in order to measure the impact of large-scale notification

Impact on Patching

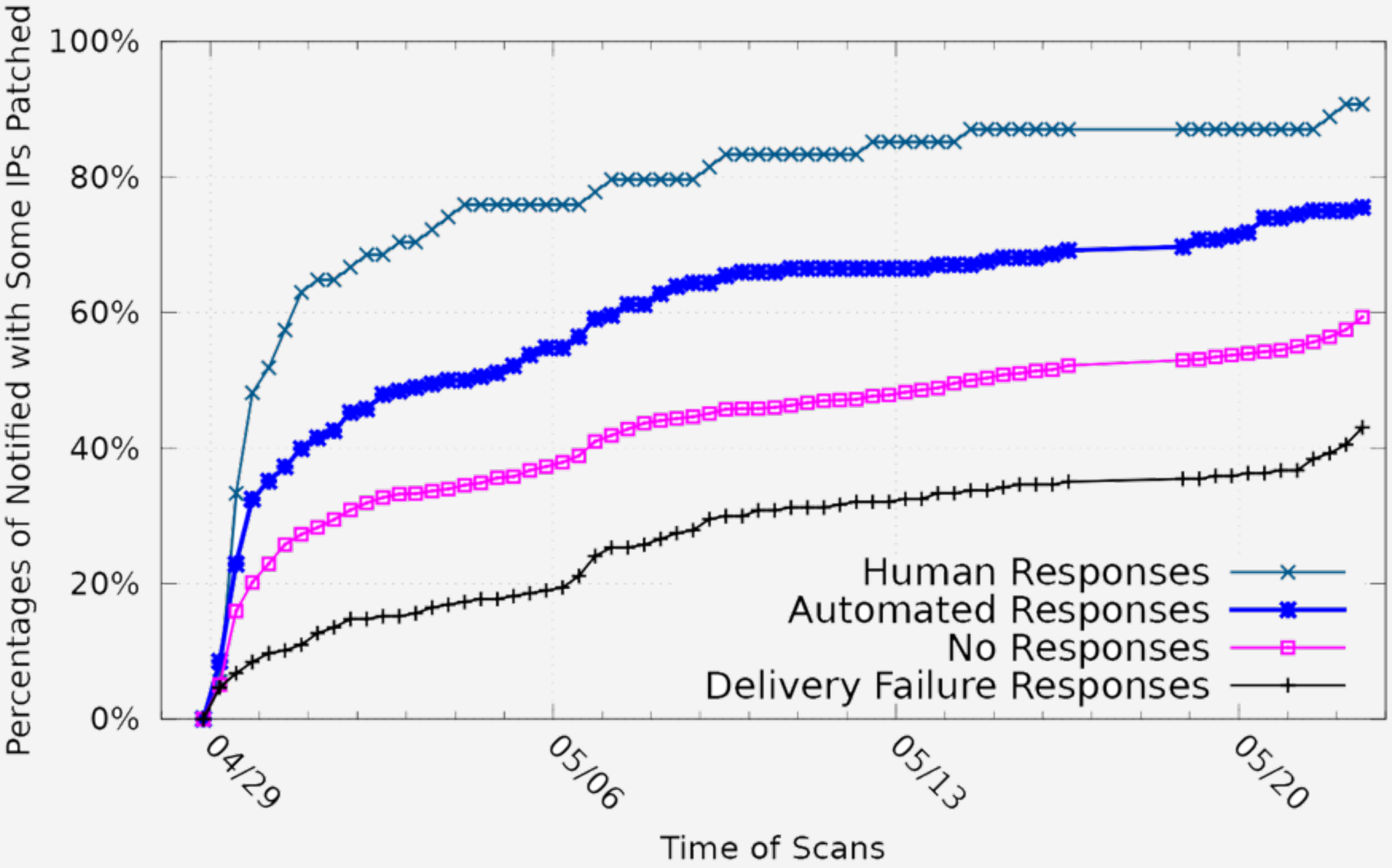


Impact on Patching



47% increase in patching

Notification Responses

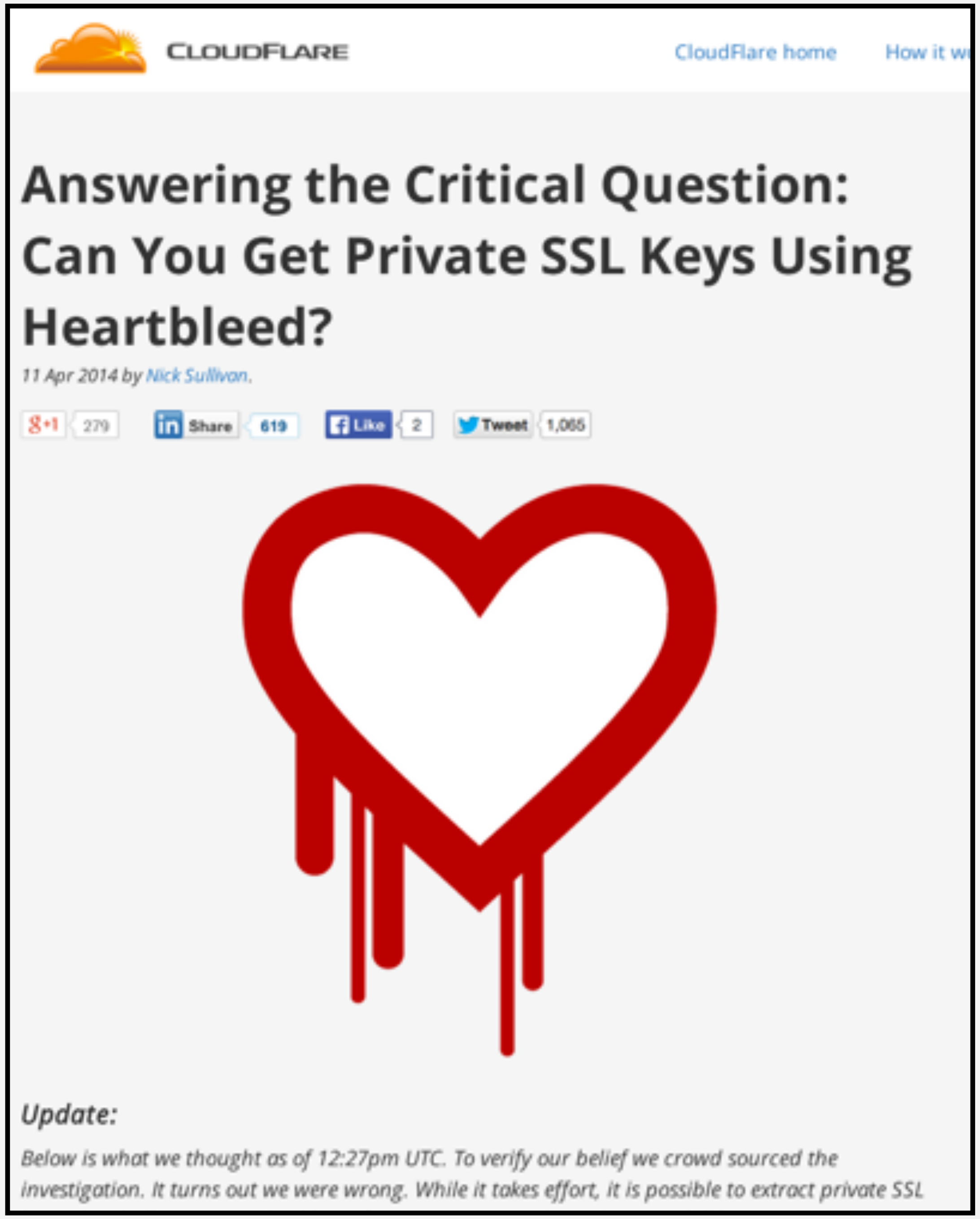


Cryptographic Keys at Risk

Patching isn’t enough—cryptographic keys can also be stolen

Proven during CloudFlare Challenge, in which keys were retrieved from generic nginx server

Security community recommended that administrators replace keys, revoke vulnerable certificates, and deploy perfect forward secrecy



Answering the Critical Question: Can You Get Private SSL Keys Using Heartbleed?

11 Apr 2014 by Nick Sullivan.

Update:

Below is what we thought as of 12:27pm UTC. To verify our belief we crowd sourced the investigation. It turns out we were wrong. While it takes effort, it is possible to extract private SSL

Cryptographic Keys

We combined our Heartbleed scans with our daily scans of the HTTPS ecosystem and ICSI's passive Certificate Notary in order to investigate certificate replacement

10.1% of the sites we found vulnerable replaced their certificates

14% *re-used* the vulnerable private key on new certificate

4% revoked their vulnerable certificates

Only 44% of connections use *Perfect Forward Secrecy*—Heartbleed did not spur further deployment

Conclusion

Heartbleed took the Internet by surprise in April 2014

Internet-scale scanning allowed us to track who was vulnerable and understand what happened

For the most part, users did well at patching, but clearly not well enough to outpace attackers and hosts remain vulnerable today

We completed a large-scale notification effort in order to help spur patching — surprisingly positive result

Ultimately, we hope that this understanding will help us be better prepared the next time this happens

Questions?

Zakir Durumeric, Frank Li, James Kasten, Johanna Amann, Jethro Beekman, Mathias Payer, Nicolas Weaver, David Adrian, Vern Paxson, Michael Bailey, J. Alex Halderman

heartbleed@umich.edu