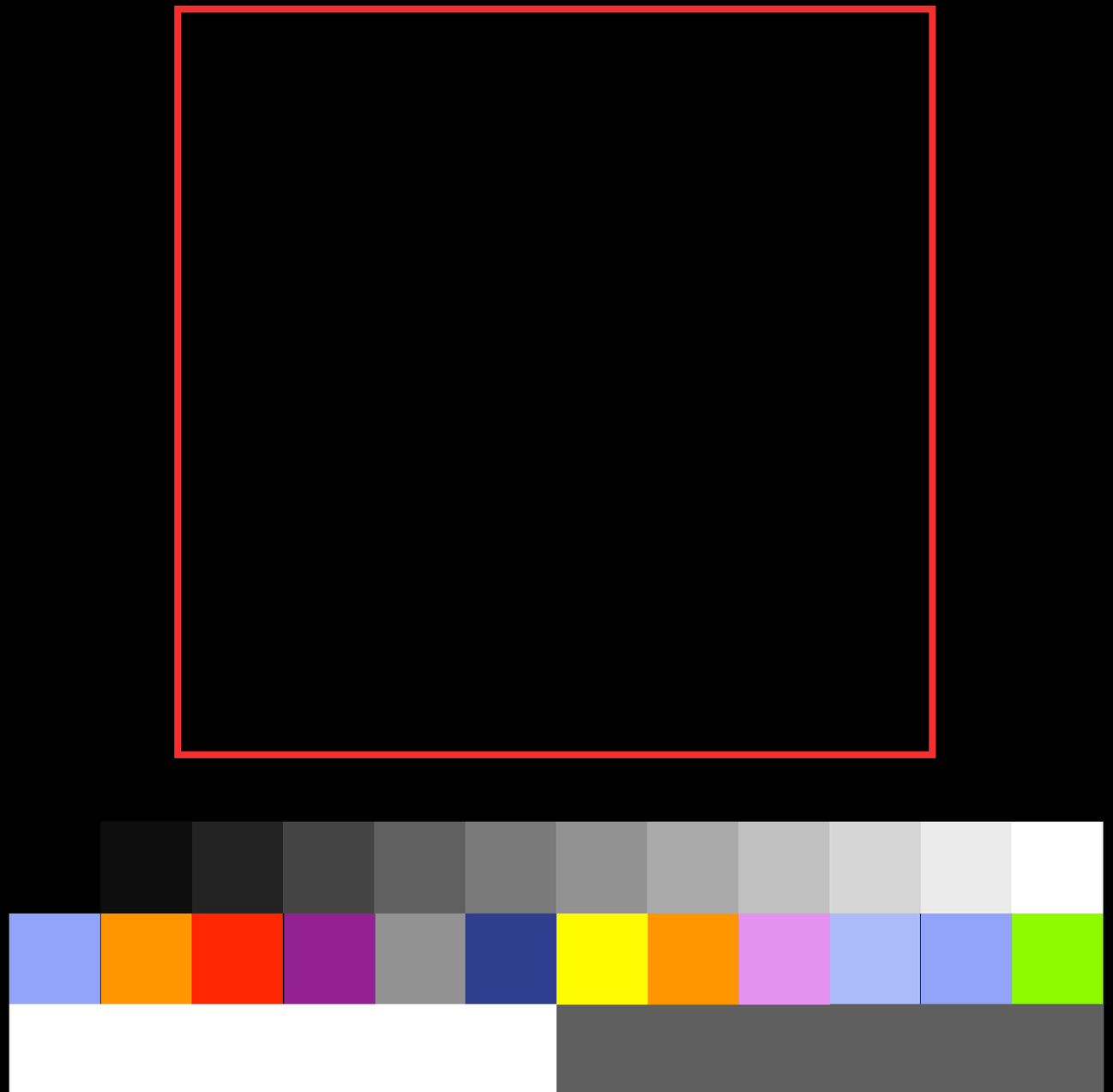


Comp[redacted]: Introduction to Security

[redacted]

[redacted]

Copyright © 2019, [redacted] All rights reserved.

Watch your email

You should get a short email with a link for the “security experiment”

Feel free to read the assignment PDF at your leisure

Finish before Sunday night, 11:59pm

“Participation credit” — we’re not grading on correctness

You’ll need three hours to do the project itself

Try to clear your schedule so you can work uninterrupted

This lecture gives you what you need to participate in the experiment

The Friday lecture covers “Web 2.0”; perhaps helpful, not essential

SWI-Prolog official account
@SWI_Prolog

Follow

For halloween we're dressing up as a language with mutable state and a single class type system. We'll have destructive asisgnment and everything!

7:21 PM - 30 Oct 2018

6 Retweets **14** Likes

1 6 14

What is security?

Security 101

The word “security” doesn’t mean very much, we need better terms

***Confidentiality*: keeping things secret (security for *who* can <u>read</u> *what*)**

***Integrity*: keeping things from changing (security for *who* can <u>write</u> *what*)**

***Availability*: keeping the system up and running (vs. “denial of service”)**

Other terms that tend to come up:

Authentication: figuring out *who* is there (for making access control decisions)

Trusted computing base: all the code that needs to be “correct” for your security

- Smaller TCB = better chance at your security being strong

What exactly is “hacking”?

“Hacking”: finding a way around the security features

What exactly is “hacking”?

“Hacking”: finding a way around the security features

Somebody Just Claimed a $1 Million Bounty for Hacking the iPhone

November 2, 2015 // 01:36 PM EST

Apple devices are widely considered extremely secure and hard to hack. But as the internet adage says, everything can be hacked—even the new iPhone.

Over the weekend, somebody claimed the $1 million bounty set by the new startup Zerodium, according to its founder Chaouki Bekrar, a notorious merchant of unknown, or zero-day, vulnerabilities.

The challenge consisted of finding a way to remotely jailbreak a new iPhone or iPad running the latest version of Apple’s mobile operating system iOS (in this case iOS 9.1 and 9.2b), allowing the attacker to install any app he or she wants with full privileges. The initial exploit, according to the terms of the challenge, had to come through Safari, Chrome, or a text or multimedia message.

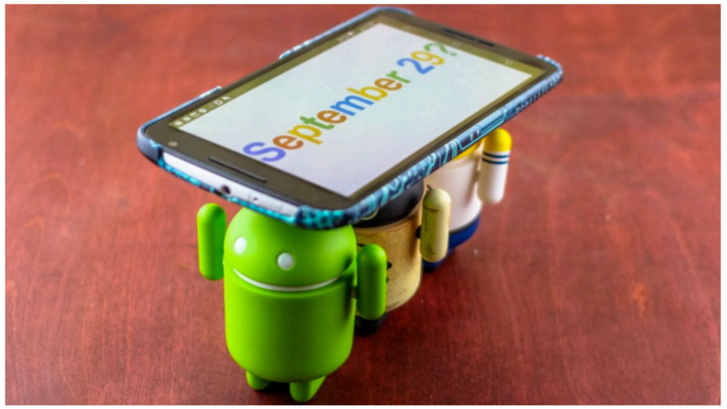
This essentially meant that a participant needed to find a series, or a chain, of unknown zero-day bugs, not just one, according to Patrick Wardle, a researcher that works at security firm Synack. For example, the Chinese white hat hacking team Pangu already found a way to jailbreak the new iPhone, but that method didn't work remotely.

What exactly is “hacking”?

Major Android Chrome exploit could make millions of phones vulnerable

By James O'Malley, 4 days ago Mobile phones

At least the good guys found the problem first



A bug has been found in the Android version of Chrome that could potentially enable hackers to install and run whatever apps they like on your phone.

The Register first reported that the bug was found in Chrome's JavaScript v8 engine, meaning that the dodgy code could conceivably be loaded onto your phone if you visit a malicious website.

Worse still, as this bug was found in one of the newest Android handsets – Google's own Nexus 6 (Project Fi version) – it suggests the problem could affect lots of phones. Which is bad.

The good news, however, is that the bug wasn't discovered by evil hackers but by one of the good guys. Chinese hacker Guang Gong showcased the exploit at the MobilePwn2Own part of the PacSec conference in Tokyo – a meeting of security experts who show off what they've discovered for the kudos.

In fact, Gong could even be in line for a cash reward from Google for finding the problem, as part of its Android Security Rewards Program, so there's no need for him to become a super villain.

way around the security features

Somebody Just Claimed a $1 Million Bounty for Hacking the iPhone

November 2, 2015 // 01:36 PM EST

Apple devices are widely considered extremely secure and hard to hack. But as the internet adage says, everything can be hacked—even the new iPhone.

Over the weekend, somebody claimed the $1 million bounty set by the new startup Zerodium, according to its founder Chaouki Bekrar, a notorious merchant of unknown, or zero-day, vulnerabilities.

The challenge consisted of finding a way to remotely jailbreak a new iPhone or iPad running the latest version of Apple’s mobile operating system iOS (in this case iOS 9.1 and 9.2b), allowing the attacker to install any app he or she wants with full privileges. The initial exploit, according to the terms of the challenge, had to come through Safari, Chrome, or a text or multimedia message.

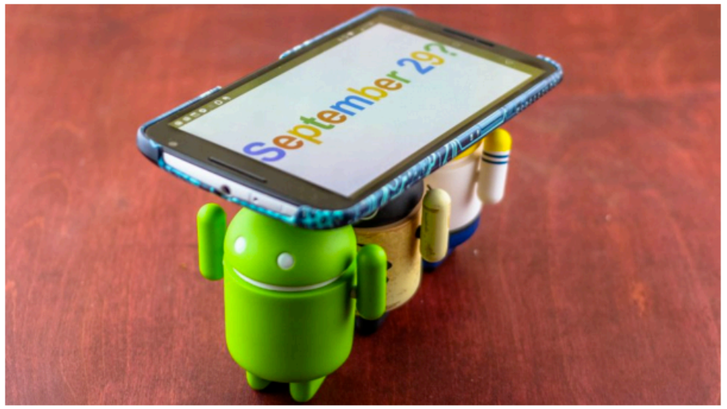
This essentially meant that a participant needed to find a series, or a chain, of unknown zero-day bugs, not just one, according to Patrick Wardle, a researcher that works at security firm Synack. For example, the Chinese white hat hacking team Pangu already found a way to jailbreak the new iPhone, but that method didn't work remotely.

What exactly w features?

Major Android Chrome exploit could make millions of phones vulnerable

By James O'Malley, 4 days ago Mobile phones

At least the good guys found the problem first



A bug has been found in the Android version of Chrome that could potentially enable hackers to install and run whatever apps they like on your phone.

The Register first reported that the bug was found in Chrome's JavaScript v8 engine, meaning that the dodgy code could conceivably be loaded onto your phone if you visit a malicious website.

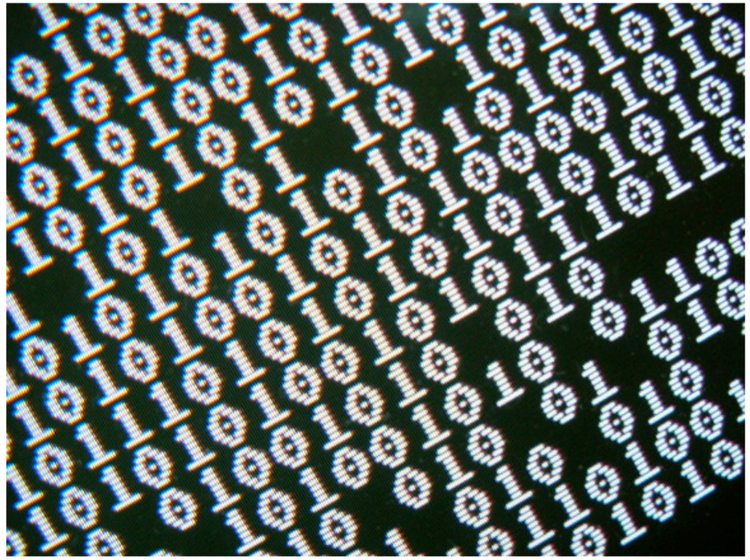
Worse still, as this bug was found in one of the newest Android handsets – Google's own Nexus 6 (Project Fi version) – it suggests the problem could affect lots of phones. Which is bad.

The good news, however, is that the bug wasn't discovered by evil hackers but by one of the good guys. Chinese hacker Guang Gong showcased the exploit at the MobilePwn2Own part of the PacSec conference in Tokyo – a meeting of security experts who show off what they've discovered for the kudos.

In fact, Gong could even be in line for a cash reward from Google for finding the problem, as part of its Android Security Rewards Program, so there's no need for him to become a super villain.

Thousands of Java applications vulnerable to nine-month-old remote code execution exploit

A flaw in the Apache Commons Collections component puts Java application servers at risk of compromise



Credit: IDGNS

COMMENTS

Lucian Constantin
IDG News Service Nov 12, 2015 4:34 AM

A popular Java library has a serious vulnerability, discovered over nine months ago, that continues to put thousands of Java applications and servers at risk of remote code execution attacks.

The flaw is located in Apache Commons, a library that contains a widely used set of Java components maintained by the Apache Software Foundation. The library is used by default in multiple Java application servers and other products including Oracle WebLogic, IBM WebSphere, JBoss, Jenkins and OpenNMS.

The flaw is specifically in the Collections component of Apache Commons and stems from unsafe deserialization of Java objects. In programming languages, serialization is the process of converting data to a binary format for storing it in a file or memory, or for sending it over the network. Deserialization is the reverse of that process.

Somebody Just Claimed a $1 Million Bounty for Hacking the iPhone

November 2, 2015 // 01:36 PM EST

Apple devices are widely considered extremely secure and hard to hack. But as the internet adage says, everything can be hacked—even the new iPhone.

Over the weekend, somebody claimed the $1 million bounty set by the new startup Zerodium, according to its founder Chaouki Bekrar, a notorious merchant of unknown, or zero-day, vulnerabilities.

The challenge consisted of finding a way to remotely jailbreak a new iPhone or iPad running the latest version of Apple's mobile operating system iOS (in this case iOS 9.1 and 9.2b), allowing the attacker to install any app he or she wants with full privileges. The initial exploit, according to the terms of the challenge, had to come through Safari, Chrome, or a text or multimedia message.

This essentially meant that a participant needed to find a series, or a chain, of unknown zero-day bugs, not just one, according to Patrick Wardle, a researcher that works at security firm Synack. For example, the Chinese white hat hacking team Pangu already found a way to jailbreak the new iPhone, but that method didn't work remotely.

Who are the players in this game?

Different attacker "types":

0: Script kiddies (i.e., no skills but can operate tools that others make)

1: Smart, motivated people with time on their hands (i.e., you)

2: Industrial research labs (i.e., you, plus more experience, plus a big budget)

3: Nation-state military/intel (i.e., research labs with fewer legal constraints)

Many other players:

Vendors, sometimes offering "bugs bounties" (worth $10k's and more)

"Zero-day" brokers (offering even more money than the vendors)

Penetration testing consultancies ("pay us, we attack you!")

Software static analysis tooling ("find your security bugs automatically!")

- Plus "formal" analysis systems ("prove your system is secure!")

Firewalls, intrusion detection systems ("build walls around your systems!")

“Is iOS more safe than Android?”

We don’t know how to measure security; we can’t answer this question!

But we can compare security features across these systems

	iOS	**Android**
App isolation	Unix process-based	Unix process-based
App permissions	Requested at time of use (originally, too many were granted by default)	Originally requested at install time, now also requested at time of use
Malware detection	In the app store (mysterious, undefined process)	In the app store (mysterious, undefined process)
Likelihood of apps or system having exploitable bugs	Higher than you’d think (Objective-C has buffer overflow issues)	Lower than you’d think (Java is mostly safe against buffer overflows)

Old vs. new software

Bugs are always in there. Somewhere.

But when discovered, they need to be fixed.
Who fixes them?

Apple: Fixes iOS and distributes patches.
iOS 13+ runs on iPhone 6S and later.
Very good adoption rates.

Android: Google distributes fixes, OEMs or carriers might not care.
Many security-critical features are auto-updated by Google, regardless.
Google-branded devices (Nexus, Pixel, etc.) are supported just like Apple iOS.
Google (finally) requiring OEMs to ship regular security patches.

Security is a process, not a destination

Microsoft, Once Infested With Security Flaws, Does an About-Face

By NICK WINGFIELD NOV. 17, 2015

Brad Smith, Microsoft's president and chief legal officer. Evan McGlinn for The New York Times

Email

Save

REDMOND, Wash. — Microsoft was once the epitome of everything wrong with security in technology. Its products were so infested with vulnerabilities that the company's co-founder, Bill Gates, once ordered all of Microsoft engineers to stop writing new code for a month and focus on fixing the bugs in software they had already built.

But in recent years, Microsoft has cleaned up its act, even impressing security specialists like Mikko Hypponen, the chief research officer for F-Secure, a Finnish security company, who used to cringe at Microsoft's practices.

Security is a process, not a destination

Microsoft, Once Infested With Security Flaws, Does an About-Face

By NICK WINGFIELD NOV. 17, 2015

Brad Smith, Microsoft's president and chief legal officer. Evan McGlinn for The New York Times

Email

Save

REDMOND, Wash. — Microsoft was once the epitome of everything wrong with security in technology. Its products were so infested with vulnerabilities that the company's co-founder, Bill Gates, once ordered all of Microsoft engineers to stop writing new code for a month and focus on fixing the bugs in software they had already built.

But in recent years, Microsoft has cleaned up its act, even impressing security specialists like Mikko Hypponen, the chief research officer for F-Secure, a Finnish security company, who used to cringe at Microsoft's practices.

"It's kind of like going to the gym every day," said Mr. Nadella, who himself runs about three miles a day. "You can't say I'm serious about security without exercising the regimen of keeping security top of mind every second, every hour of the day."

...

Microsoft estimates that it now spends more than $1 billion a year on security-related initiatives, including acquisitions. It acquired three security start-ups in the last year alone, and the number of security employees at the company increased 20 percent during that time.

Ethics matter!

Example: You're an NSA researcher and you discover a remote exploit for Android phones. What do you do?

Option 1: Report it to Google because it could impact U.S. citizens!

Option 2: Weaponize it so you can listen in on foreign leaders.

Ethics matter!

Example: You're an NSA researcher and you discover a remote exploit for Android phones. What do you do?

Option 1: Report it to Google because it could impact U.S. citizens!

Option 2: Weaponize it so you can listen in on foreign leaders.

File Is Said to Confirm N.S.A. Spied on Merkel

By MELISSA EDDY JULY 1, 2015

Email

Save

BERLIN — More than three years ago, Chancellor Angela Merkel "professed to be at a loss" over the option of debt relief that would best stem the crisis in Greece. Yet even then, she feared that no amount of coaching from visiting experts would resolve the problems in Athens.

The chancellor's musings over whether it would be better to allow private investors to incur losses on their bond holdings or to create a transfer union within the eurozone come from a protocol or memorandum dated 2011 that was said to have been intercepted by the United States' National Security Agency.

Ms. Merkel's conversation with an unidentified adviser, along with another document from the chancellor's top aide for European affairs and a list of 69 telephone numbers said to belong to members of the German government and their aides, were made public Wednesday by the anti-secrecy group WikiLeaks.

NSA: Ethical? Rational? Legal?

Ethics: You need a framework to help decide what's okay.

Hippocratic oath: "Do no harm!"

- But harms are relative. There's no clear right answer.
- Consider "quarantines": what happens to the infected people?

Rationality: This is where things get interesting.

The NSA has an "offensive" and a "defensive" mission.

They "need" vulnerabilities for the offensive missions.

"Weapons" have limited shelf-life. Single-use.

Legality: *Way* beyond the scope of Comp[redacted].

Ethics for college students

For you, there's zero ambiguity about this

You cannot attack anything other than yourself

Totally cool: Attack your own phone, attack your own browser, etc.

Somewhat cool: Attack your own Gmail account

- If you figure out how to "steal" credentials from yourself and then reuse them, that's fine
- If you want to probe at Gmail and break into the whole thing, that's going to land you in jail

Entirely uncool: Attack Canvas, change your grades

Standard operating procedure: "responsible disclosure"

You first notify the vendor and given them an "appropriate" chance to respond

You then are free to go public and describe how it worked

- But if it's going to take the vendor years to fix it, you might withhold critical details

Common vulnerability patterns

How about our RPN calculator?

Here's the JavaScript that adds text to the screen

```
function printParagraph(text) {
    var textBox = $("#textOutput")
    textBox.append("<p>" + text + "</p>") // cross-site scripting opportunity!
    textBox.scrollTop(textBox.prop("scrollHeight")); // scroll to the bottom
}
```

Q: *What happens if the "text" has JavaScript in it?*

A: Remote code injection

If "untrusted" content can make it back to the user (e.g., if this was Twitter or Facebook or whatever), then this could be exploitable

"Cross-site scripting" (XSS) attack

Live-coding demo time: XSS

What can XSS do?

Social networking: force users to post something, like something, etc.
Instant viral spreading!

“Phishing”: pop up a dialog, “you need to enter your password again”
Sometimes this happens anyway, so users won’t find it suspicious

When the attacker gets to run arbitrary code (JavaScript or otherwise), they can violate confidentiality, integrity, and availability.

How do you fix XSS?

Option 1: Escaping / filtering

The **j2html** library can do this - arbitrary text in a tag will be safely escaped

Putting it together - make sure that raw user input can't flow into the HTML

Option 2: Set rules in the browser that ban JavaScript

Mozilla's "Content Security Policies" (now supported everywhere)

- Bans all inline JavaScript
- Bans JavaScript's "eval" and other ways of generating dynamic code
- Rules about hosts JavaScript will be allowed from (i.e., only trusted web servers)

Lots of nitty gritty details to do this right

- https://www.owasp.org/index.php/XSS_(Cross_Site_Scripting)_Prevention_Cheat_Sheet

More fun: cross-site request forgery (CSRF)

Let’s say you want to force me to buy your stuff on Amazon

What if there was an Amazon link to do it? One-click purchasing!

Attacker gets me to visit the attacker’s home page, forces a GET request

```
<img src="http://amazon.com/buy/my/thing/">
```

If you’re logged into Amazon, this will pass your cookies along, perform a HTTP GET request

More fun: cross-site request forgery (CSRF)

Let's say you want to force me to
What if there was an Amazon link

Attacker gets me to visit the atta
request

```
<img src="http://amazon.
```

If you're logged into Amazon, thi
perform a HTTP GET request

This is your "project" for this week. We'll give you an "insecure" server and you'll try to fix it. Start today, finish by Sunday.

"Participation grade": It's okay if you don't succeed. Please give it your best effort. Different students have different versions of the problem.

How do you fix CSRF?

Option 1: GET requests should have no side effects (no mutation!)

It's much harder for an attacker to cause a PUT or POST request

Option 2: Secret tokens the attacker can't see

- *Not* cookies!
- "Hidden form fields"
- Something in the URL
- Something in a JSON request

Again, many subtleties to getting this right

- https://www.owasp.org/index.php/CSRF_Prevention_Cheat_Sheet

Ever wonder why your bank's website logs you out so fast?

- They're paranoid about this class of attack and they're proactively trying to defeat it.

How do you fix CSRF?

Option 1: GET requests should have no side effects (no mutation!)

It's much harder for an attacker to cause a PUT or POST request

Option 2: Secret tokens the attacker can't see

- *Not* cookies!
- "Hidden form fields"
- Something in the URL
- Something in a JSON request

Attacker can generate GET request with

`<img src="…">`

Possible from *any* web site.

Cookies are sent along!

Again, many subtleties to getting this right

- https://www.owasp.org/index.php/CSRF_Prevention_Cheat_Sheet

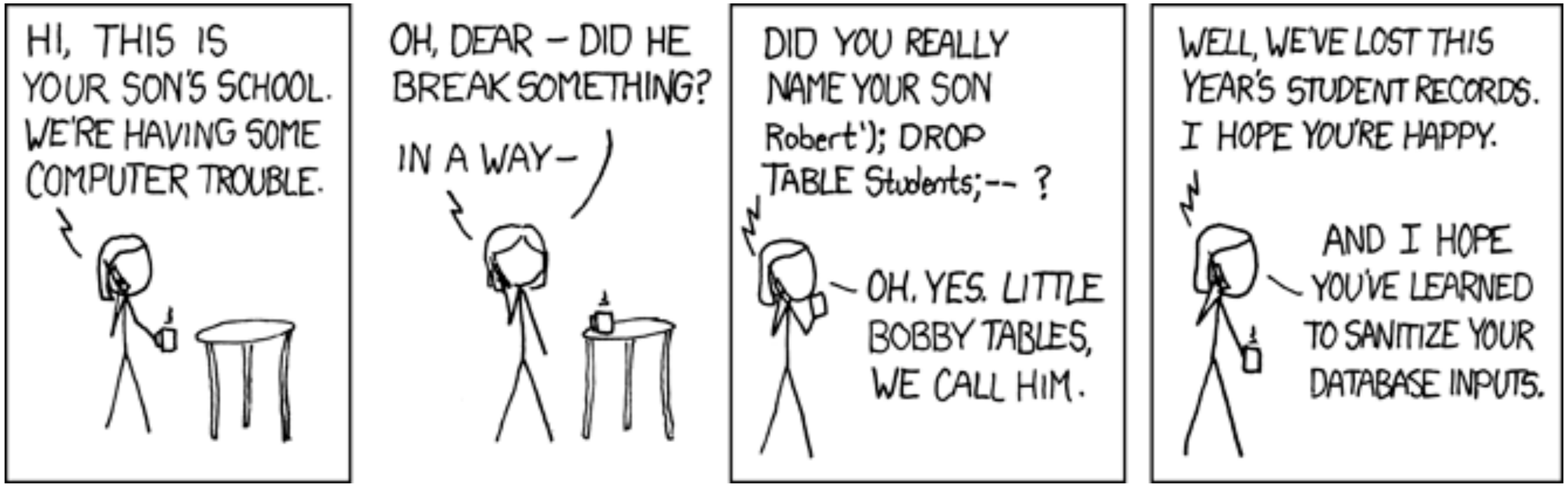
Ever wonder why your bank's website logs you out so fast?

- They're paranoid about this class of attack and they're proactively trying to defeat it.

SQL injection attacks?

Similar to XSS: untrusted data being fed into a SQL (database) query

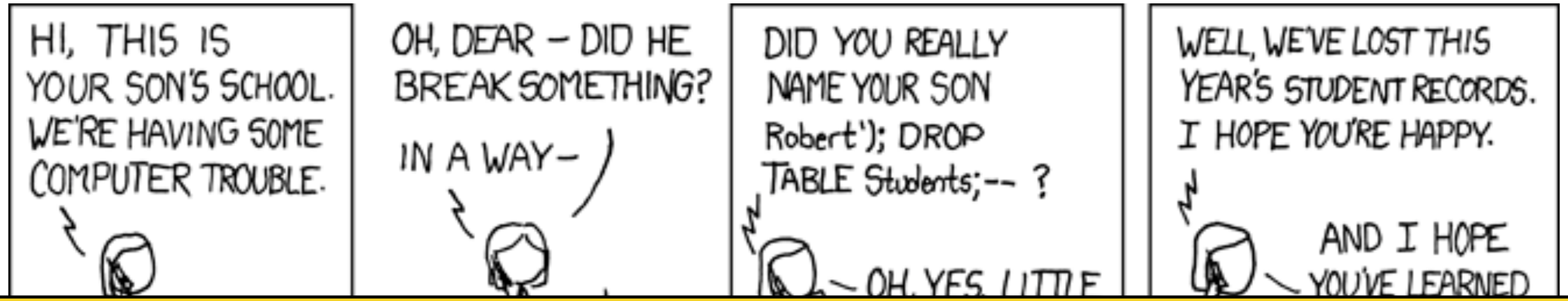
Similar solutions: use a library to help you build your SQL query ("prepared statements"); don't just concatenate strings together.



SQL injection attacks?

Similar to XSS: untrusted data being fed into a SQL (database) query

Similar solutions: use a library to help you build your SQL query ("prepared statements"); don't just concatenate strings together.



Fun, interactive SQL injection attack tutorial (assumes no knowledge of SQL, but does assume basic JavaScript knowledge)
https://www.executeprogram.com/course/sql/lesson/basic-tables

Buffer overflow attacks?

Programs in C and C++ use arrays in a dumb way

It's not a crash-bug to index beyond the end of an array! It works!

You'll learn more about how call stacks work in Comp[redacted], ...

In short: you can write binary code to the stack then force a jump to it.

How do you fix it?

Very expensive static analysis tools (Coverity, Fortify, etc.)

Don't use C or C++ (Java programs don't have this class of vulnerability)

More advanced tricks

- fancy compilers, address space layout randomization, and much much more

More advanced stuff?

Security researchers keep inventing new attacks

Maybe it’s a new target to go after (e.g., self-driving cars)

They also keep inventing new general-purpose technologies

E.g.: “Two factor authentication” - getting fancier than passwords

And new ways of dealing with problems

E.g., Running in a virtual machine, to isolate the malware

Sometimes: disagreement about what’s even an attack

The Register®
Biting the hand that feeds IT

Security

Stealth web crypto-cash miner Coin Hive back to the drawing board as blockers move in

We've got something much more ethical anyway, say devs

By Iain Thomson in San Francisco 19 Oct 2017 at 07:02 19 SHARE ▼

Malwarebytes has had enough of Coin Hive's alt-currency-generating browser-side code, and is now automatically blocking it.

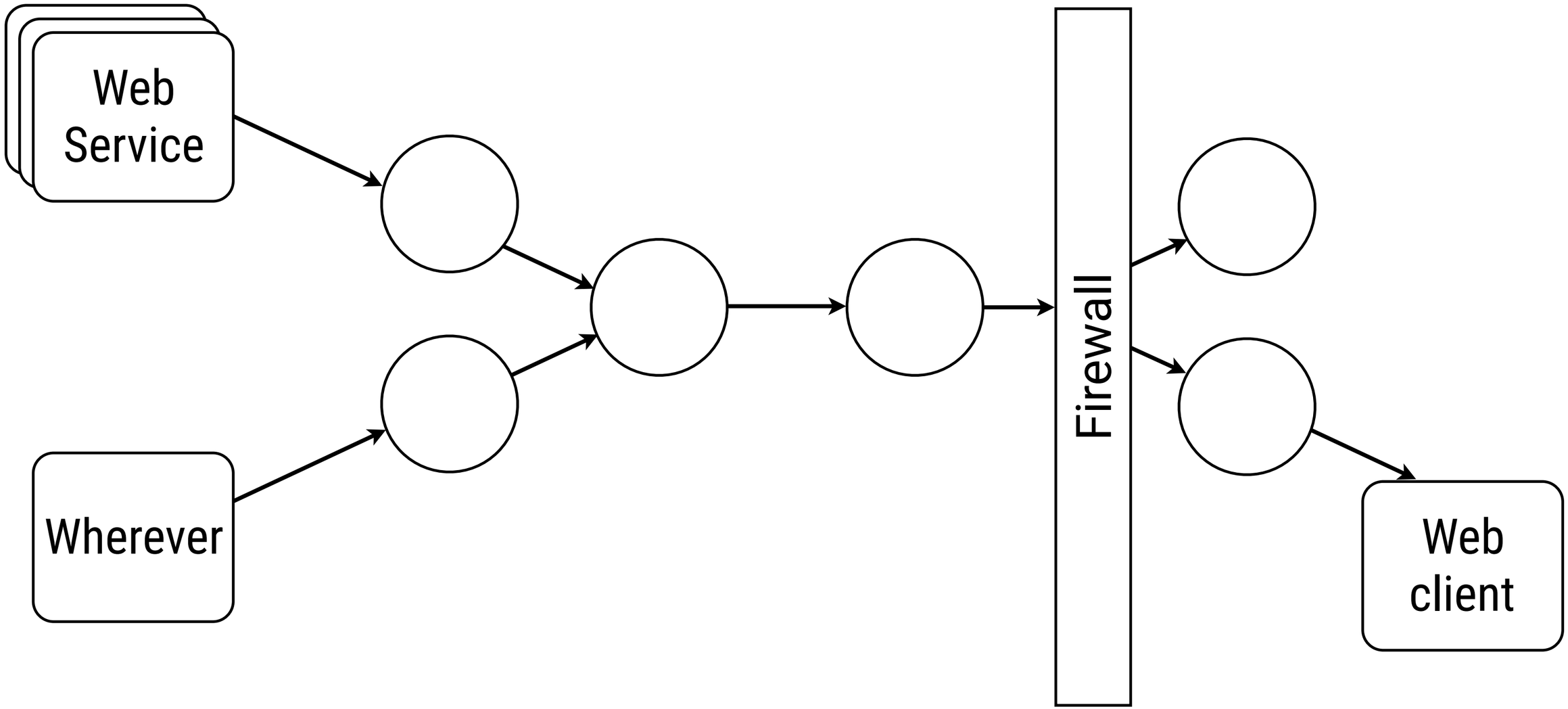
The biz joins ad-block plugins in preventing Coin Hive's Monero-crafting JavaScript from running in webpages, using visitors' electricity and hardware to mine new money. Coin Hive is a legit outfit, and its mining code is supposed to be embedded in pages to earn site owners' revenue as an alternative to annoying ads. However, this freely available tool has been abused.

Malwarebytes said coinhive.com, which hosts the mining software, was the second-most-commonly blocked domain by its users, with 130 million users expressing their disdain for the technology.

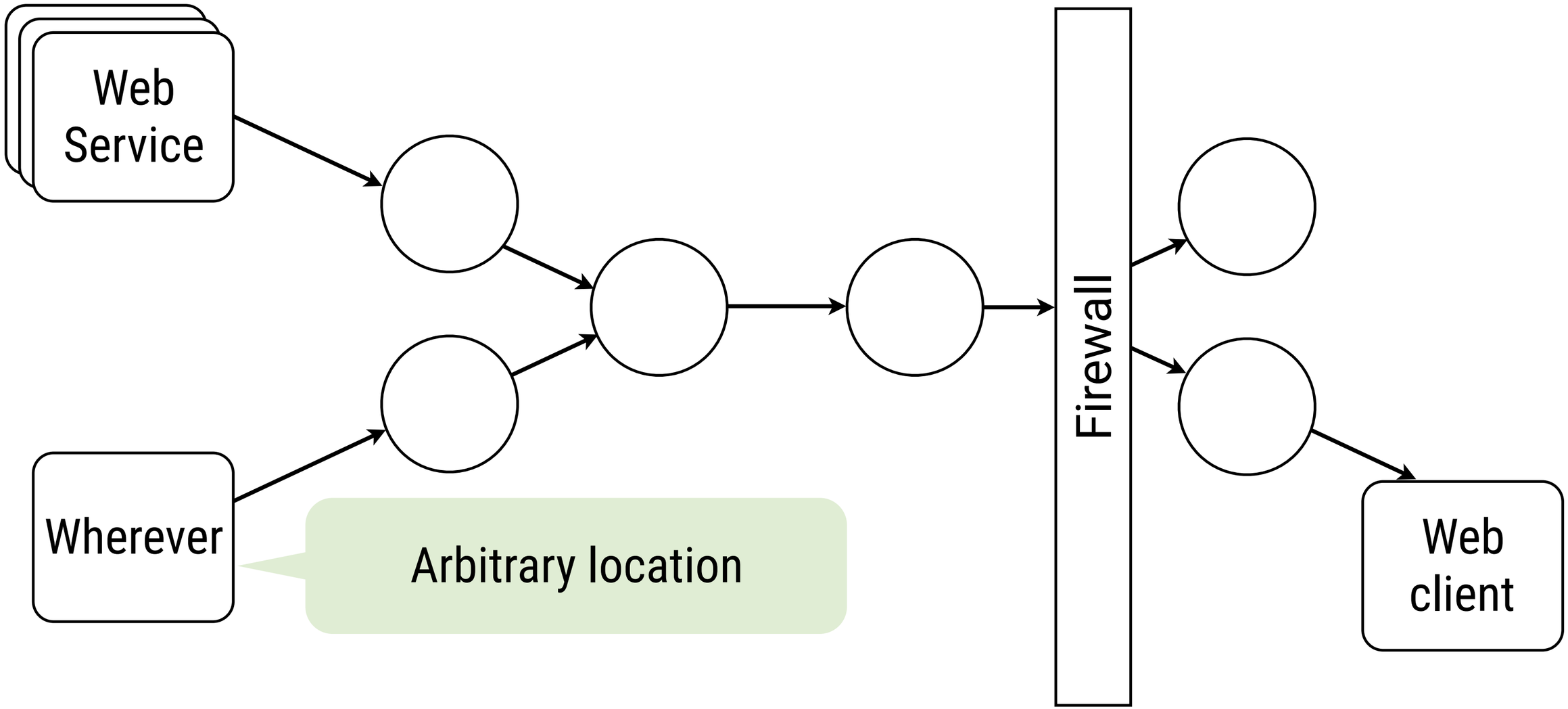
"We do not claim that Coin Hive is malicious, or even necessarily a bad idea," noted Adam Kujawa, director of Malwarebytes Labs, on

Where's your attacker?

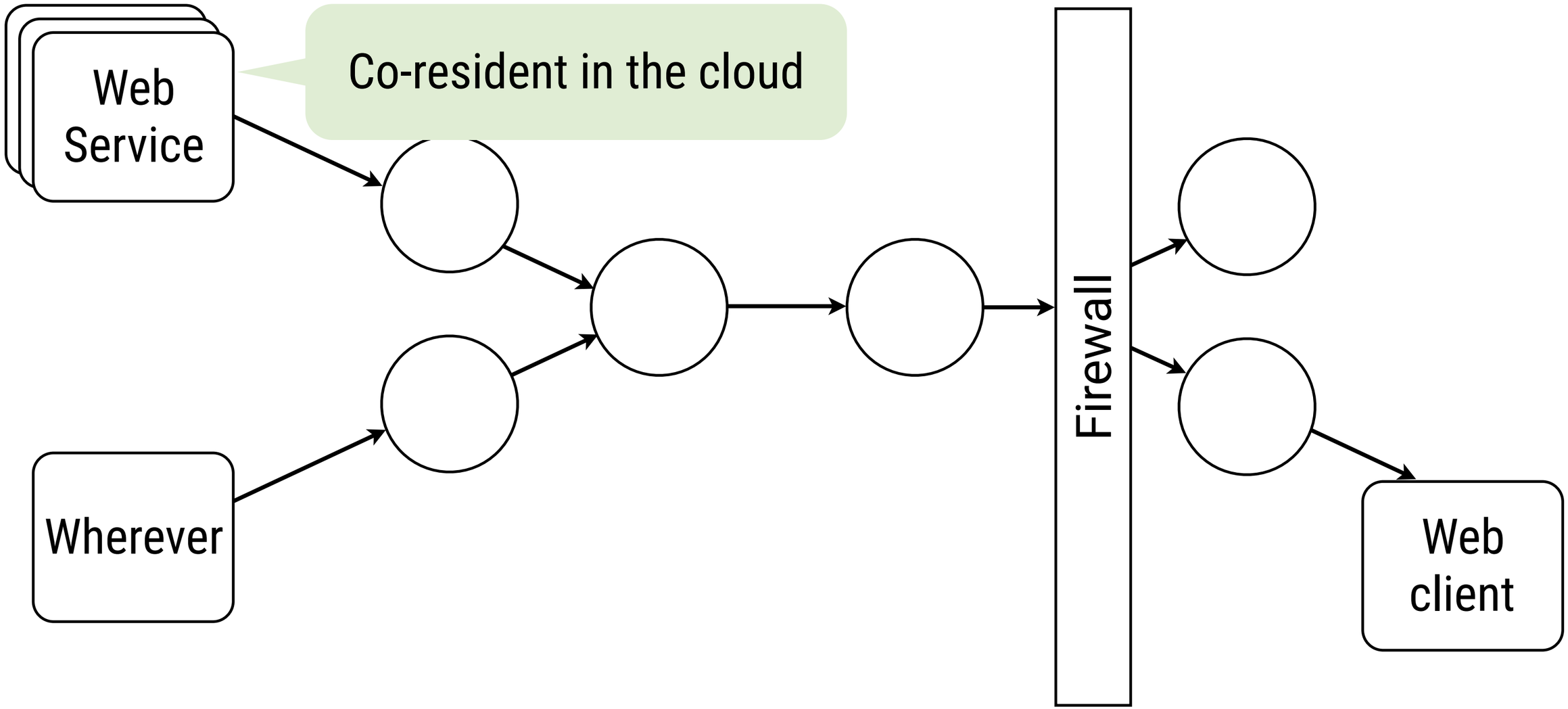
The attacker’s location matters



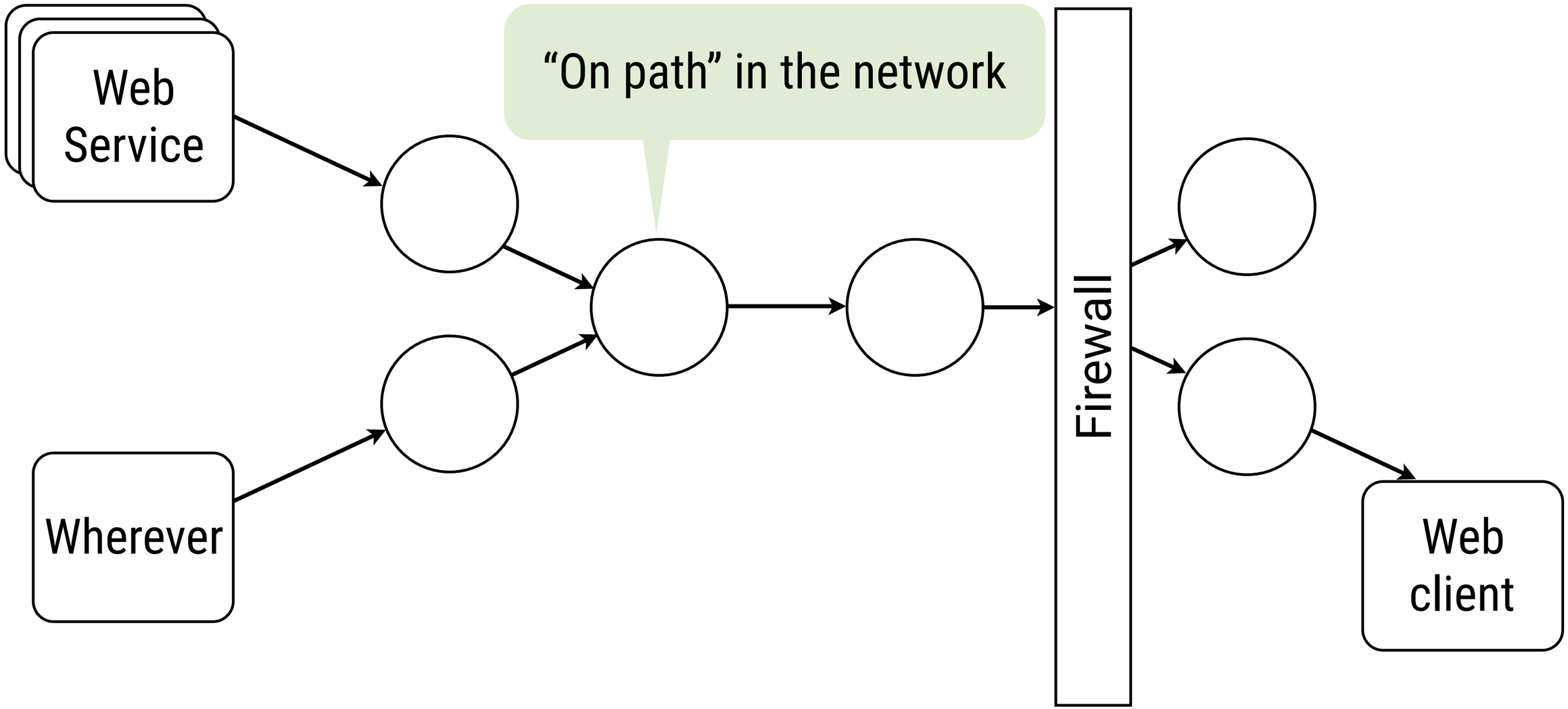
The attacker’s location matters



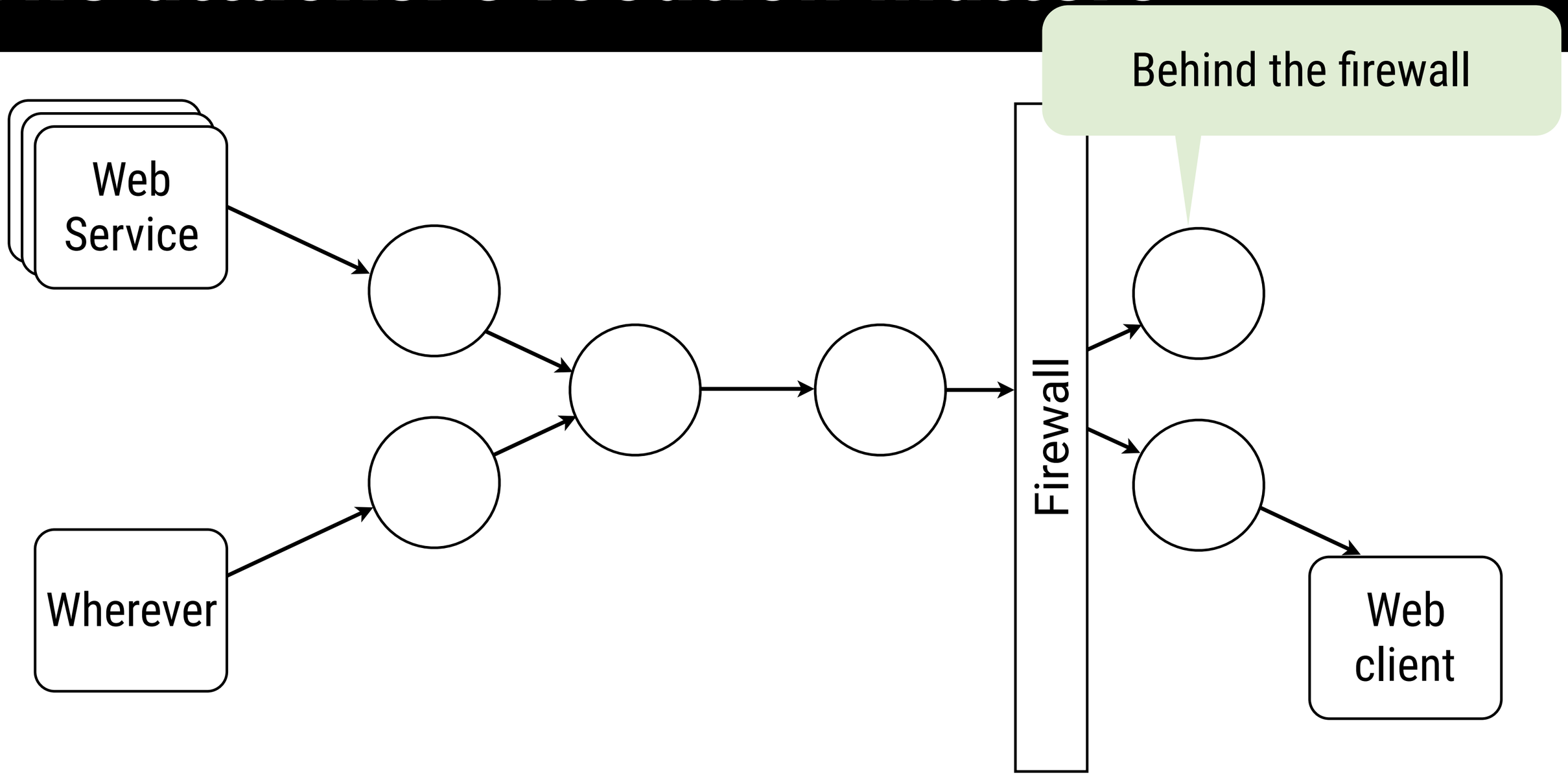
The attacker’s location matters



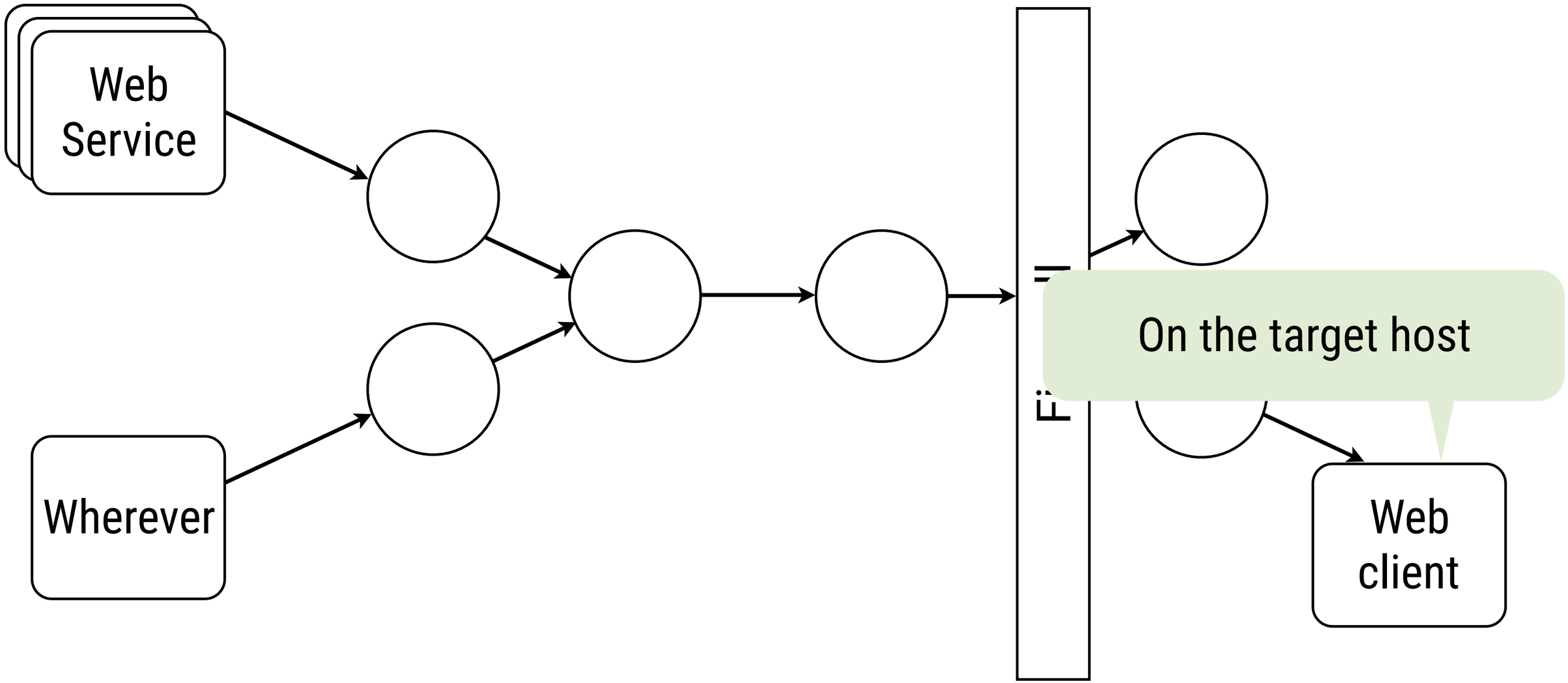
The attacker's location matters



The attacker's location matters



The attacker’s location matters



Examples where location matters

Co-resident (client or server)

Precise timing of network connections can leak crypto private keys!

- Taking large numbers to large powers takes a variable amount of time if you do it the obvious way.
- Recent attacks take advantage of “speculative execution” performance side-effects (Meltdown, Spectre, others).
- No “noise” from the network if you’re on the same machine.

On-path (network)

You can do a “man-in-the-middle” attack on the communications

- Passive surveillance
- Active blocking (e.g., China’s “Great Firewall”)
- “Racing” fraudulent responses back to the client

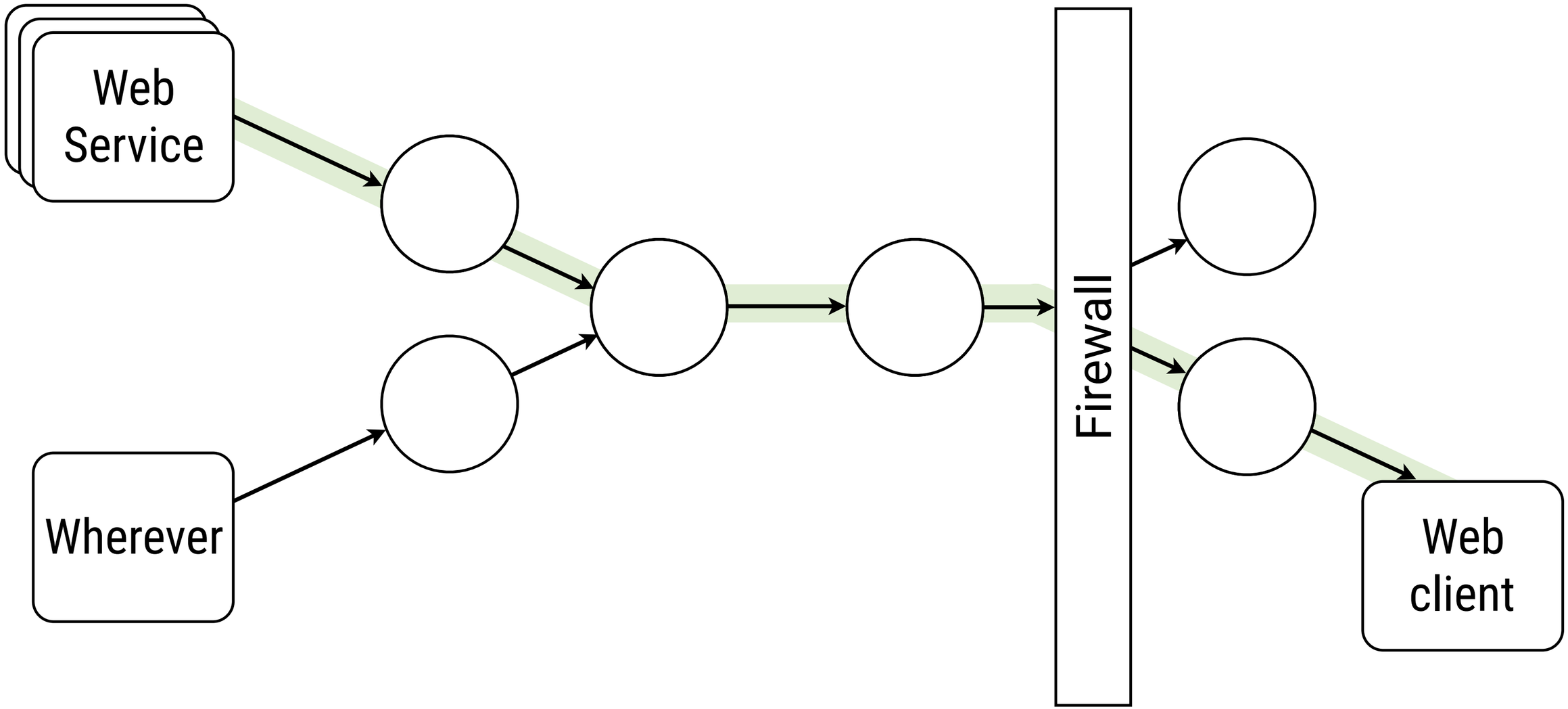
Behind the firewall

You can see open services that might not be meant for the public

- E.g., All Android apps have full network access; do you want an Android phone on your WiFi network?

More fun if you’re on the same WiFi segment: listen to all packets!

How does crypto help?



Crypto (ideal)

Passive man-in-the-middle adversaries learn nothing

Active man-in-the-middle adversaries can drop or repeat traffic

But they never *learn* anything beyond the network addresses of the endpoints.

The most they can accomplish is killing the connection, or *traffic analysis*.

Web client (with `https`) gains strong proof of the server's identity

Web server learns *nothing* about the client; typically do passwords, etc.

Crypto (ideal)

Passive man-in-the-middle adversaries le

Active man-in-the-middle adversaries can

But they never *learn* anything beyond the ne endpoints.

The most they can accomplish is killing the

Web client (with *`https`*) gains strong proo

Web server learns *nothing* about the client; t

The New York Review of Books

‘We Kill People Based on Metadata’

David Cole

Rick Bowmer/AP Photo

The National Security Agency’s $1.5 billion data storage facility in Bluffdale, Utah, June 2013

Supporters of the National Security Agency inevitably defend its sweeping collection of phone and Internet records on the ground that it is only collecting so-called “metadata”—who you call, when you call, how long you talk. Since this does not include the actual content of the communications, the threat to privacy is said to be negligible. That argument is profoundly misleading.

Of course knowing the content of a call can be crucial to establishing a particular threat. But metadata alone can provide an extremely detailed picture of a person’s most intimate associations and interests, and it’s actually much easier as a technological matter to search huge amounts of metadata than to listen to millions of phone calls. As NSA General Counsel Stewart Baker has said, “metadata absolutely tells you everything about somebody’s life. If you have enough metadata, you don’t really need content.” When I quoted Baker at a recent debate at Johns Hopkins University, my opponent, General Michael Hayden, former director of the NSA and the CIA, called Baker’s comment “absolutely correct,” and raised him one, asserting, “We kill people based on metadata.”

It is precisely this power to collect our metadata that has prompted one of Congress’s most bipartisan initiatives in recent years. On May 7, the House Judiciary Committee voted 32-0 to adopt an amended form of the USA Freedom Act, a bill to rein in NSA spying on Americans, initially proposed by Democratic Senator Patrick Leahy and Republican Congressman James Sensenbrenner. On May 8, the House Intelligence Committee, which has until now opposed any real reform of the NSA, also unanimously approved the same bill. And the Obama administration has welcomed the development.

For some, no doubt, the very fact that this bill has attracted such broad bipartisan approval will be grounds for suspicion. After all, this is the same Congress that repeatedly reauthorized the 2001 USA Patriot Act, a law that was also proposed by Sensenbrenner and on which the bulk collection of metadata was said to rest—even if many members of Congress were not aware of how the NSA was using (or abusing) it. And this is the same administration that retained the NSA’s data collection program, inherited from its predecessor, as long as it was a secret, and only called for reform when the American people learned from the disclosures of NSA contractor Edward Snowden that the government was routinely collecting phone and Internet records on all of us. So, one might well ask, if Congress and the White House, Republicans and Democrats, liberals and conservatives, all now agree on reform, how meaningful can the reform be?

This is a reasonable question. This compromise bill addresses only one part of the NSA’s surveillance activities, and does not do nearly enough to address the many other privacy-invasive practices that we now know the NSA has undertaken. But it’s nonetheless an important first step, and would introduce several crucial reforms affecting all Americans.

First, and most importantly, it would significantly limit the collection of phone metadata and other “business records.” Until now, the NSA and the Foreign Intelligence Surveillance Court have aggressively interpreted a USA Patriot Act provision that authorized collection of business records “relevant” to a counterterrorism investigation. The NSA convinced the court that because it might be useful in the future to search through anyone’s calling history to see if that person had been in contact with a suspected terrorist, the agency should be able to collect everyone’s records and store them for five years.

The NSA has said it only searched its vast database of our calling records when it had reasonable suspicion that a phone number was connected to terrorism. But it did not have to demonstrate the basis for this suspicion to a judge. Moreover, it was authorized to collect data on all callers one, two, or three steps removed from the suspect number—an authority that can quickly generate more than one million phone numbers of innocent Americans from a single suspect

Crypto (in practice)

SSL (secure socket layer) / TLS (transport layer security) dates to 1995

Lots of features to negotiate which specific cryptosystems, etc.

Version rollback attacks are significant

Modern practice: removing older crypto algorithms altogether

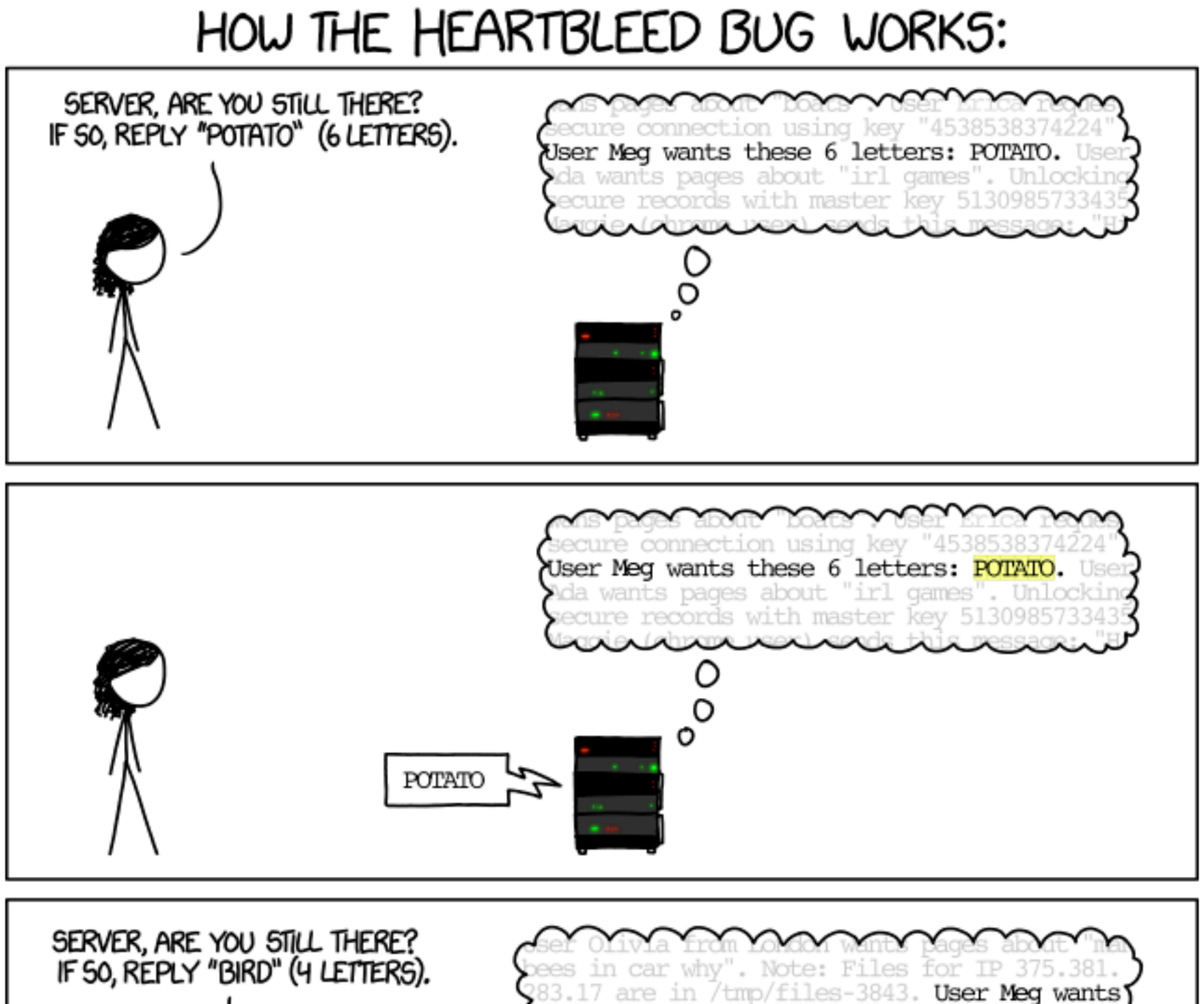
- Especially “export modes” which were deliberately weakened cryptosystems
- The latest (TLS 1.3) has formal mathematical proofs of crypto correctness!

Dangerous implementation bugs (“Heartbleed” bug)

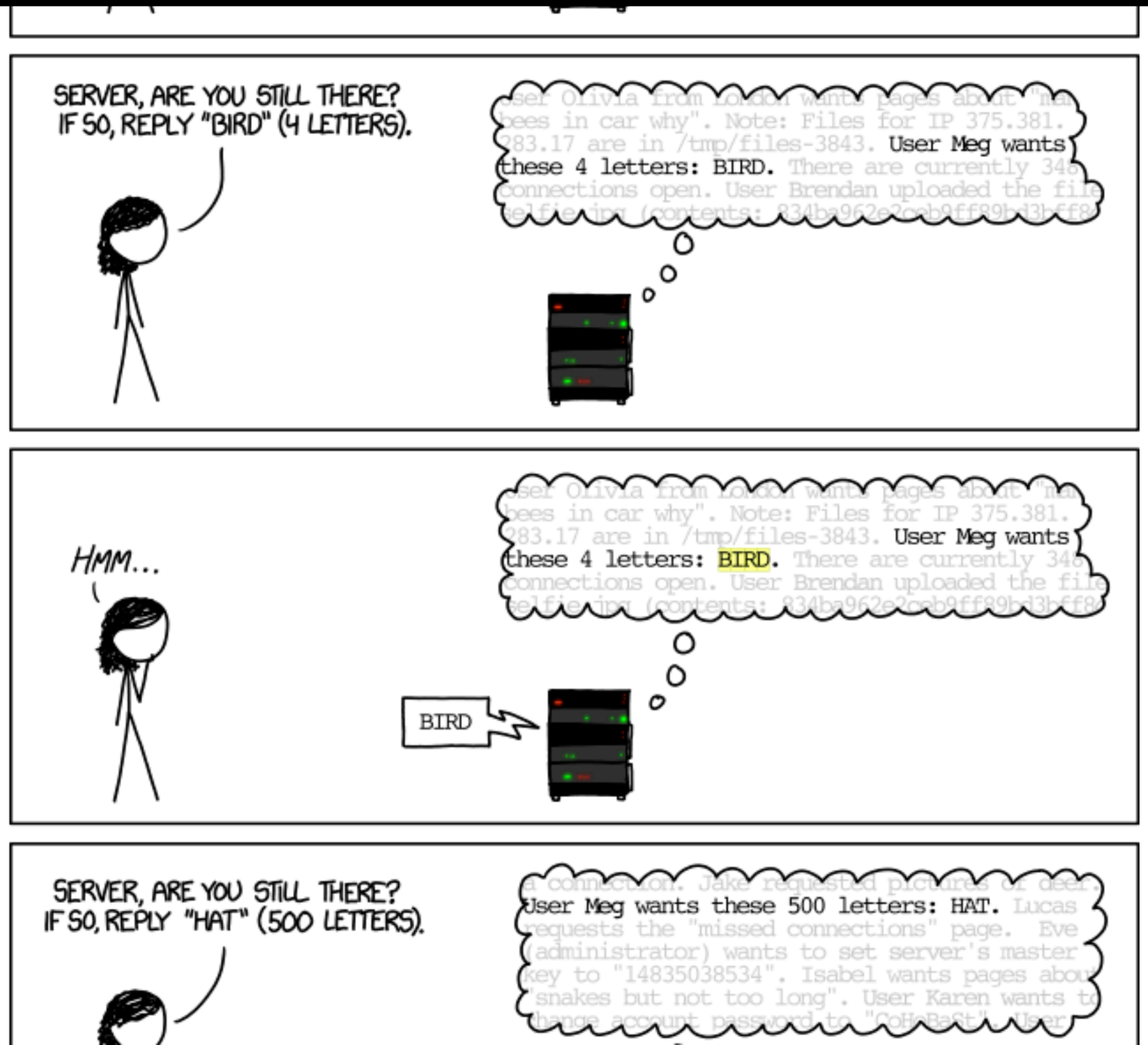
Who gets to issue the certificates?

Weird questions about what capabilities the NSA (and friends) have

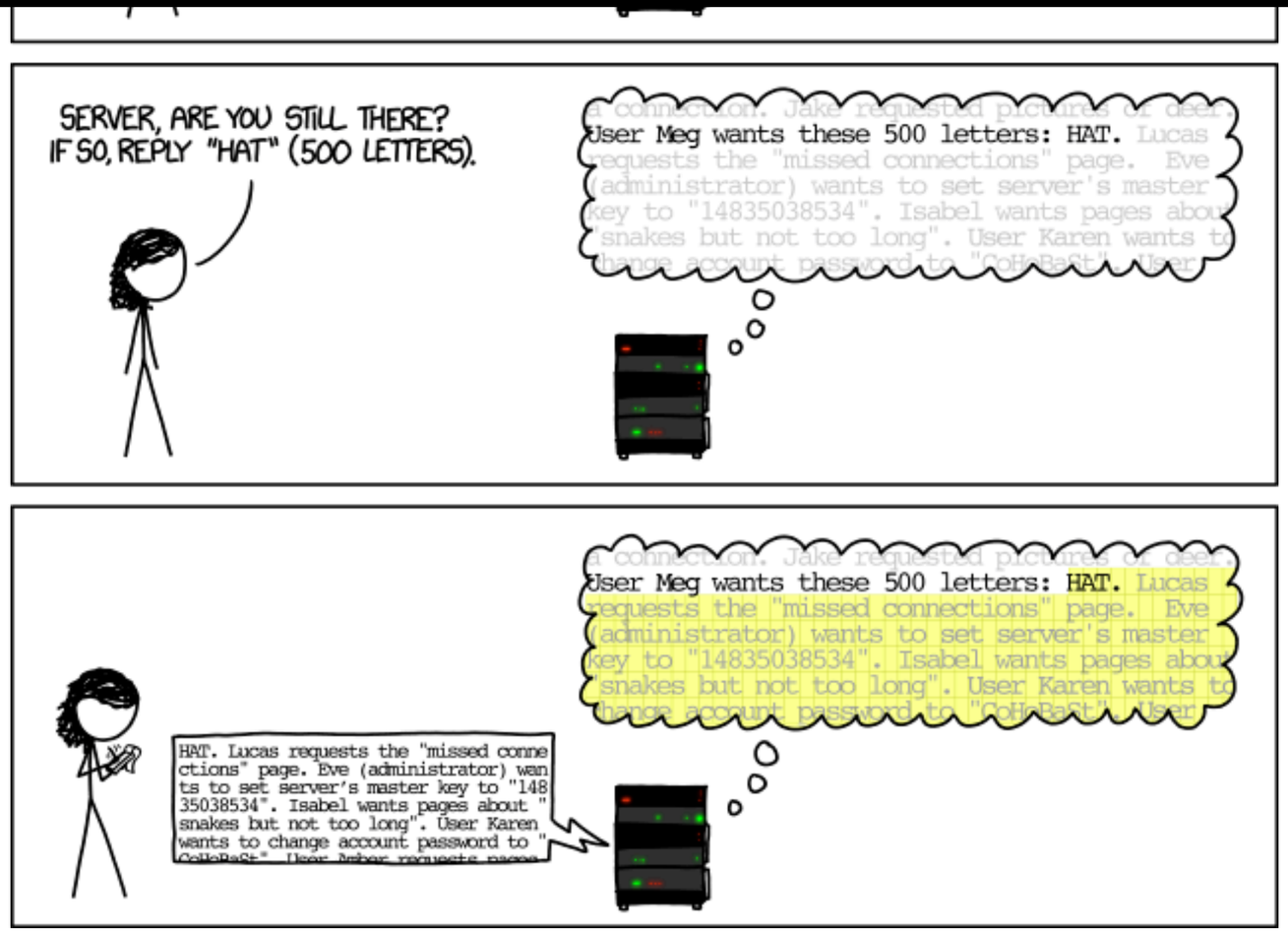
Example: OpenSSL Heartbleed bug (2014)



Example: OpenSSL Heartbleed bug (2014)



Example: OpenSSL Heartbleed bug (2014)



https://xkcd.com/1354/

Certificate authorities

Your browser trusts CAs

You visit Amazon

Amazon send you a “certificate”

CA: “this public key belongs to Amazon.com”

Your browser: I trust that CA; crypto used this key

Therefore, this must be the real Amazon!

But what if a CA is malicious?

Your browser trusts hundreds of CAs!

Certificate authorities

Your browser trusts CAs

You visit Amazon

Amazon send you a “certificate”

CA: “this public key belongs to Amazon.com”

Your browser: I trust that CA; crypto used this key

Therefore, this must be the real Amazon!

But what if a CA is malicious?

Your browser trusts hundreds of CAs!

Google Security Blog

The latest news and insights from Google on security and safety on the Internet

Distrusting WoSign and StartCom Certificates

October 31, 2016

Posted by Andrew Whalley, Chrome Security

Certificate Authorities (CAs) play a key role in web security by issuing digital certificates to website operators. These certificates are trusted by browsers to authenticate secure connections to websites. CAs who issue certificates outside the policies required by browsers and industry bodies can put the security and privacy of every web user at risk.

Google has determined that two CAs, WoSign and StartCom, have not maintained the high standards expected of CAs and will no longer be trusted by Google Chrome, in accordance with our Root Certificate Policy. This view is similar to the recent announcements by the root certificate programs of both Apple and Mozilla. The rest of this post provides background to that decision and how we plan to minimize disruption while still protecting users.

Background

On August 17, 2016, Google was notified by GitHub's security team that WoSign had issued a certificate for one of GitHub's domains without their authorization. This prompted an investigation, conducted in public as a collaboration with Mozilla and the security community, which found a number of other cases of WoSign misissuance.

Certificate authorities

Your browser trusts CAs

You visit Amazon

Amazon send you a “certificate”

CA: “this public key belongs to Amazon.com”

Your browser: I trust that CA; crypto used this k

Therefore, this must be the real Amazon!

But what if a CA is malicious?

Your browser trusts hundreds of CAs!

Google Security Blog

The latest news and insights from Google on security and safety on the Internet

The investigation concluded that **WoSign knowingly and intentionally misissued certificates in order to circumvent browser restrictions** and CA requirements… When presented with this evidence, WoSign and StartCom management actively attempted to mislead the browser community… [W]e have concluded there is a pattern of issues and incidents that indicate an approach to security that is not in concordance with the responsibilities of a publicly trusted CA.

…

Beginning with Chrome 56, **certificates issued by WoSign and StartCom after October 21, 2016 00:00:00 UTC will not be trusted**.

Fun reading: https://www.schrauger.com/the-story-of-how-wosign-gave-me-an-ssl-certificate-for-github-com

ains without their authorization. This

olic as a collaboration with Mozilla and the

of other cases of WoSign misissuance.

What are your attacker's motives?

Different attackers have different goals

“Low-value targets” vs. “high-value targets”

There’s more value in hacking Trump’s phone than your phone

If it costs $100k’s to get a previously unknown (“zero day”) vulnerability do you...

- Use it to install malware on millions of computers, indiscriminately?
- Use it to install malware on a small number of computers, because they’re important?

Most “botnet” attacks go after low-hanging fruit (e.g., unpatched Windows XP boxes), don’t use state-of-the-art methods.

Some attackers are incentivized to keep you happy!

Your machine is a platform to attack other machines.

Different attackers have

“Low-value targets” vs. “high-value targets

There’s more value in hacking Trump’s phone

If it costs $100k’s to get a previously unknow

- Use it to install malware on millions of
- Use it to install malware on a small nur important?

Most “botnet” attacks go after low-hanging fr boxes), don’t use state-of-the-art methods.

Some attackers are incentivized to keep yo

Your machine is a platform to attack other m

The New York Times

When Trump Phones Friends, the Chinese and the Russians Listen and Learn

By Matthew Rosenberg and Maggie Haberman

Oct. 24, 2018

WASHINGTON — When President Trump calls old friends on one of his iPhones to gossip, gripe or solicit their latest take on how he is doing, American intelligence reports indicate that Chinese spies are often listening — and putting to use invaluable insights into how to best work the president and affect administration policy, current and former American officials said.

Mr. Trump’s aides have repeatedly warned him that his cellphone calls are not secure, and they have told him that Russian spies are routinely eavesdropping on the calls, as well. But aides say the voluble president, who has been pressured into using his secure White House landline more often these days, has still refused to give up his iPhones. White House officials say they can only hope he refrains from discussing classified information when he is on them.

Mr. Trump’s use of his iPhones was detailed by several current and former officials, who spoke on the condition of anonymity so they could discuss classified intelligence and sensitive security arrangements. The officials said they were doing so not to undermine Mr. Trump, but out of frustration with what they considered the president’s casual approach to electronic security.

American spy agencies, the officials said, had learned that China and Russia were eavesdropping on the president’s cellphone calls from human sources inside foreign governments and intercepting communications between foreign officials.

On Thursday morning, Mr. Trump said this article was “soooo wrong!” and asserted that he used only government phones and “only one seldom used government cell phone.” He did not claim there were other factual errors in The New York Times’ reporting.

Donald J. Trump
@realDonaldTrump

The so-called experts on Trump over at the New York Times wrote a long and boring article on my cellphone usage that is so incorrect I do not have time here to correct it. I only use Government Phones, and have only one seldom used government cell phone. Story is soooo wrong!

Some attackers want to make you miserable

Several variations on this theme

You backup your computer, right?

It's all about the money, of course

One by one, we've been making it harder for them to get paid

Notice how they don't just ask for a credit card?

Nation-state attacks?

TECHNOLOGY

Internet Giants Erect Barriers to Spy Agencies

By DAVID E. SANGER and NICOLE PERLROTH JUNE 6, 2014

Google servers in Douglas County, Ga. The company is encrypting more data as it moves between servers. Connie Zhou/Google

Email

Save

MOUNTAIN VIEW, Calif. — Just down the road from Google's main campus here, engineers for the company are accelerating what has become the newest arms race in modern technology: They are making it far more difficult — and far more expensive — for the National Security Agency and the intelligence arms of other governments around the world to pierce their systems.

Nation-state attacks?

Eric Grosse, Google's security chief, suggested in an interview that the N.S.A.'s own behavior invited the new arms race.

"I am willing to help on the purely defensive side of things," he said, referring to Washington's efforts to enlist Silicon Valley in cybersecurity efforts. "But signals intercept is totally off the table," he said, referring to national intelligence gathering.

"No hard feelings, but my job is to make their job hard," he added.

TECHNOLOGY

Internet Giants Erect Barriers to Spy Agencies

By DAVID E. SANGER and NICOLE PERLROTH JUNE 6, 2014

Google servers in Douglas County, Ga. The company is encrypting more data as it moves between servers. Connie Zhou/Google

Email

Save

MOUNTAIN VIEW, Calif. — Just down the road from Google's main campus here, engineers for the company are accelerating what has become the newest arms race in modern technology: They are making it far more difficult — and far more expensive — for the National Security Agency and the intelligence arms of other governments around the world to pierce their systems.

An interesting trend…

Old school (Windows 3.x, Windows 95, even Windows XP)

User is running with full (root/admin) privileges, all the time

WinXP's non-admin accounts were virtually unusable

New school (Windows Vista and later, standard Unix and OS X)

You have to type a password to "elevate" privileges

Default apps are limited in what they can do

The latest (Android, iOS, Chromebook, others)

You *cannot* elevate your privileges. You don't have "root" on your phone.

- Requires "jailbreaking".
- Jailbroken phones sometimes are missing features (e.g., disables *Google Pay*).

Harder for you to "steal" DRM-protected video.

Harder for malware to install itself.

Harder to install an ad blocker.

Should you jailbreak to steal video if it makes your phone more vulnerable?

Future trends…

Increasingly ambiguous attackers

Nation-states demand “backdoor” access to phones, demand local servers

Advertisers (and apps) want to profile you, monetize you

Internet reaching more and more devices, which aren’t secure (yet)

SCADA (industrial control systems)

“Internet of Things” (light bulbs, etc.)

Curiosity: Whose fault if an attacker uses a bug to steal your car?

Remotely unlock the doors, start the ignition, drive away with your car!

Who’s incentivized to fix the problem? Who’s skilled to fix it?

- The manufacturer? The insurance company? The end user?
- Will manufacturers warranty their cars against electronic theft?
- Would you rather let Zipcar / Uber / somebody else deal with it?