No One In The Middle

Enabling network access control via transparent attribution

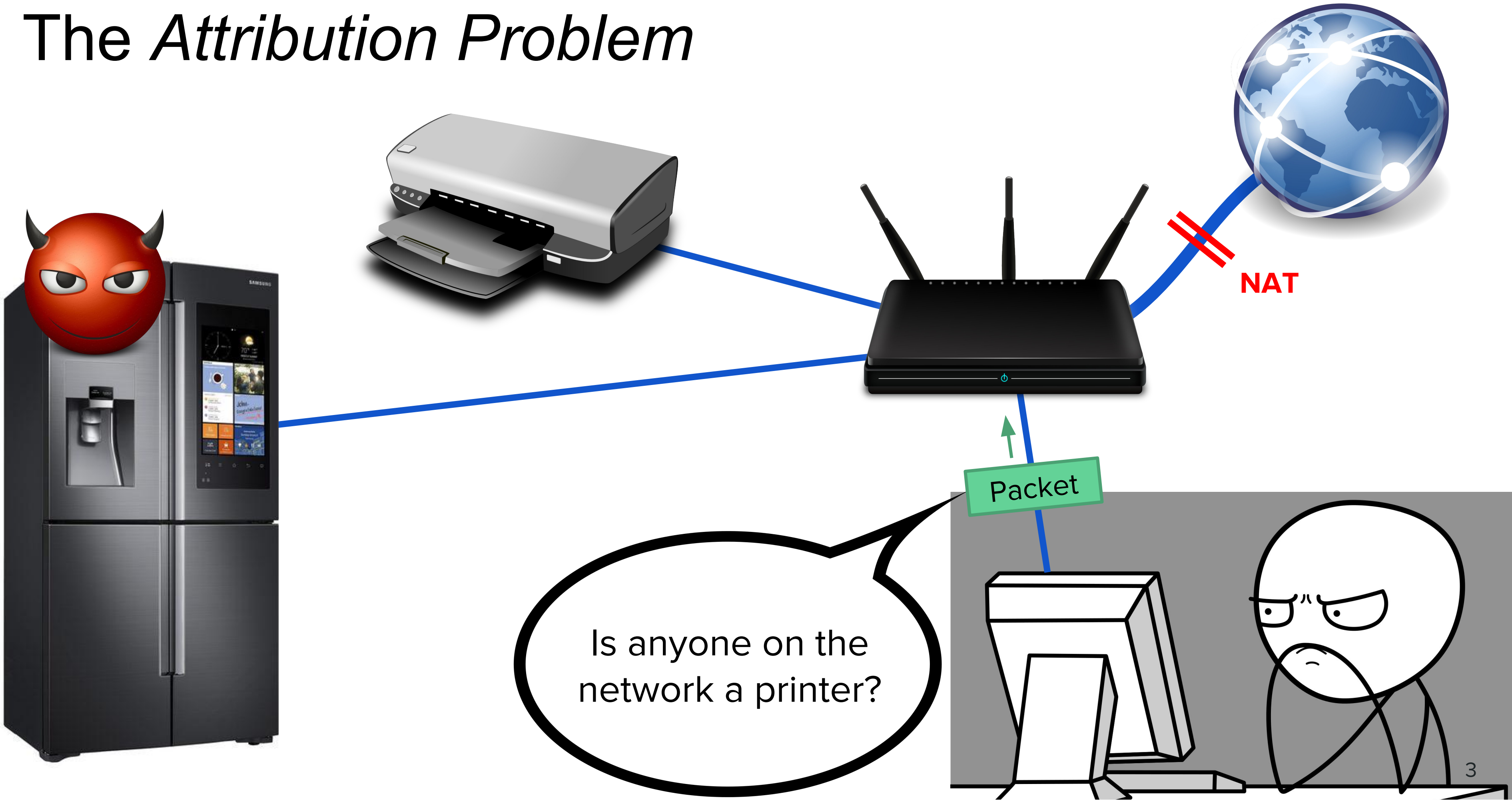
Jeremy Erickson, Qi Alfred Chen, Xiaochen Yu,
Erinjen Lin, Robert Levy, Z. Morley Mao

University of Michigan, Ann Arbor

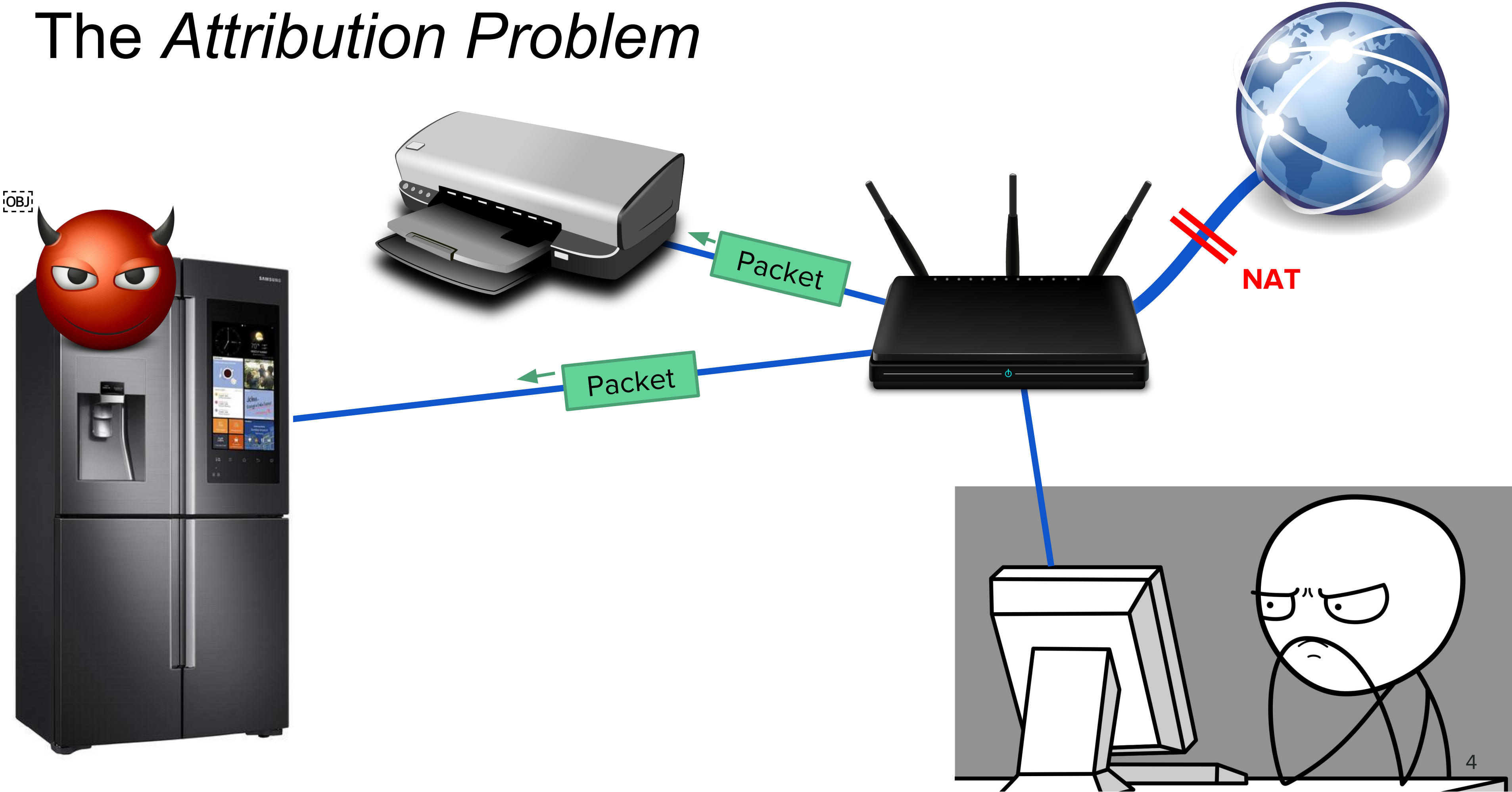
Commodity small network

SAMSUNG
nest
70

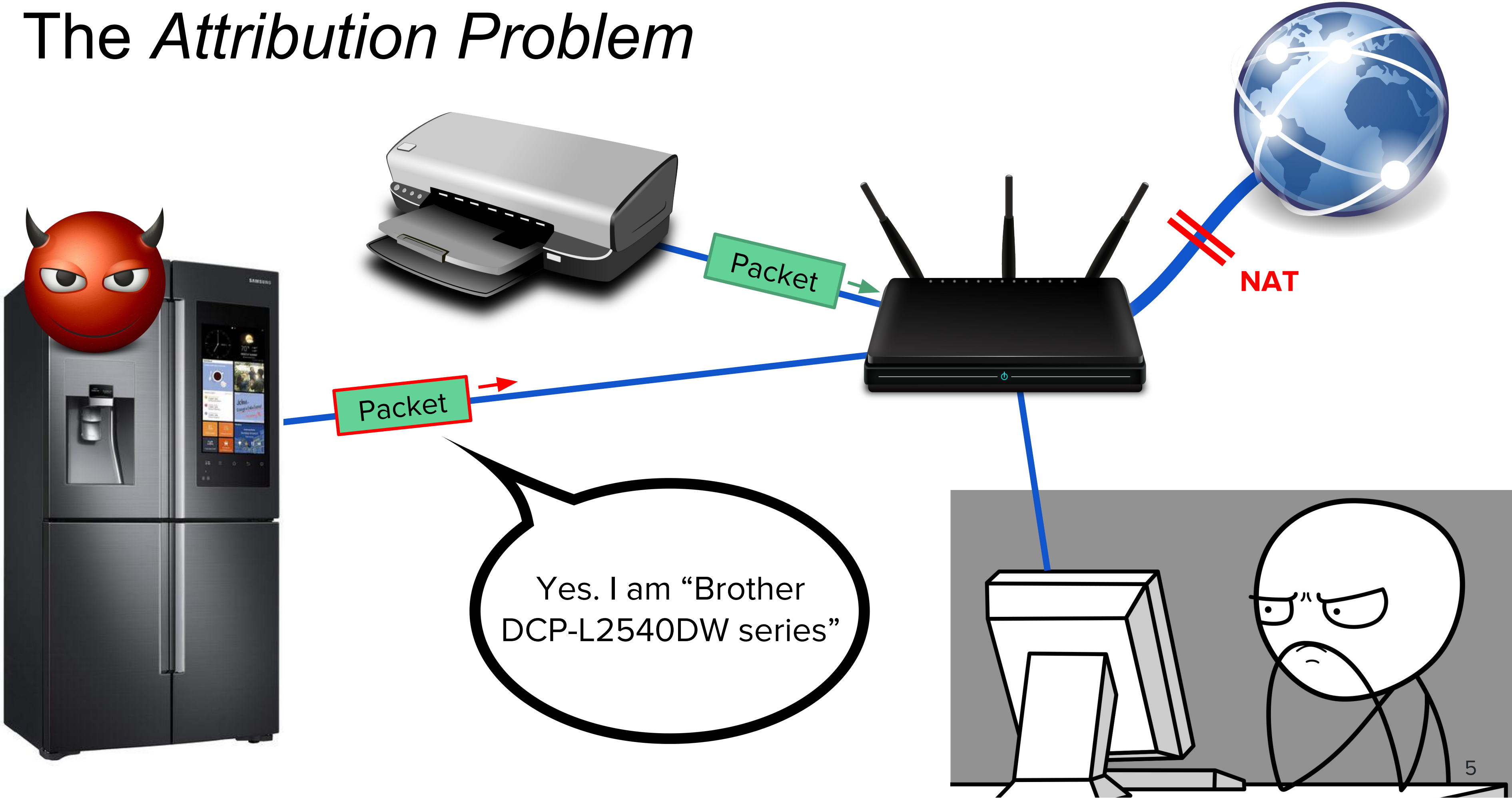
The *Attribution Problem*



The *Attribution Problem*

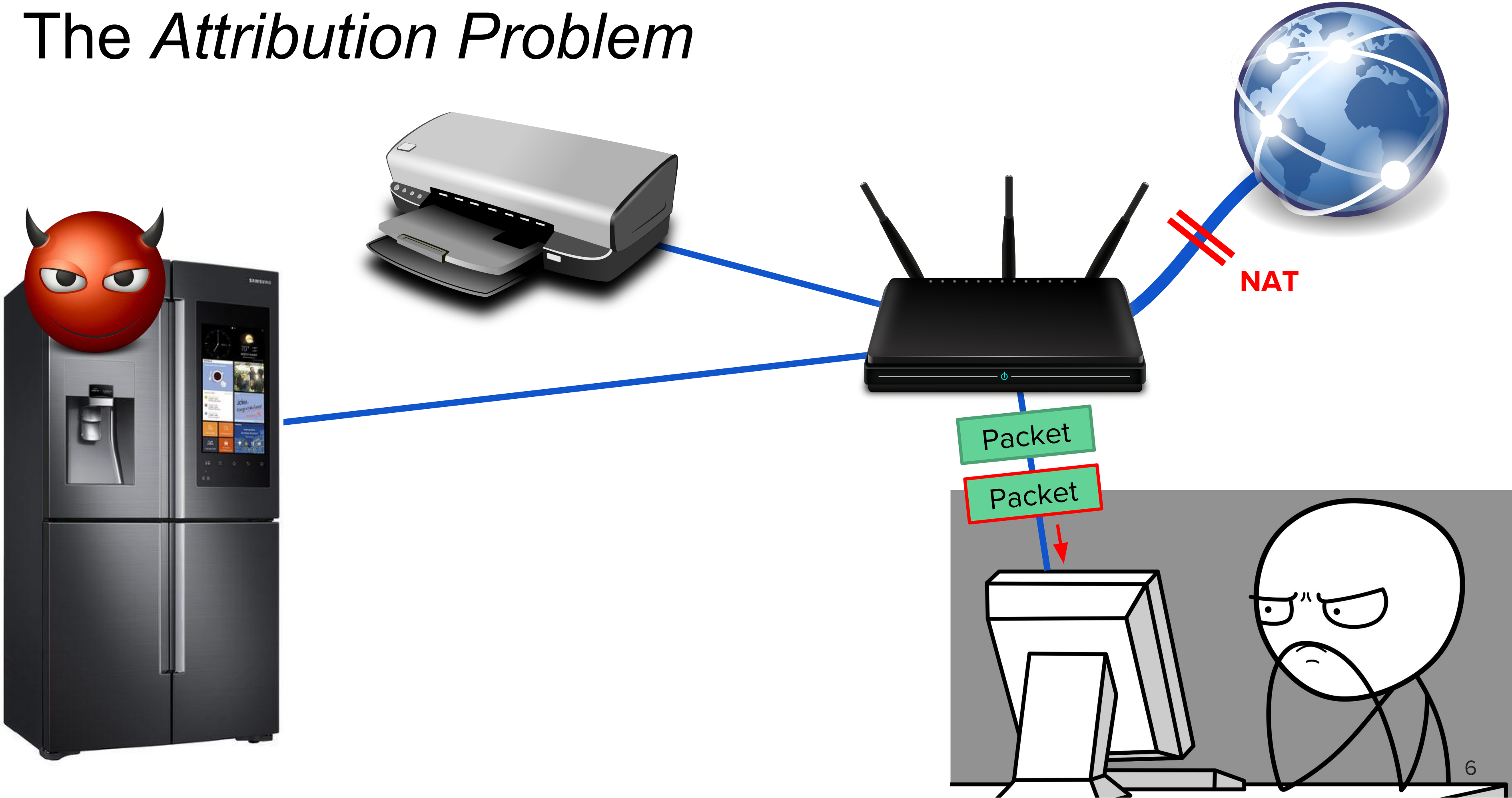


The *Attribution Problem*

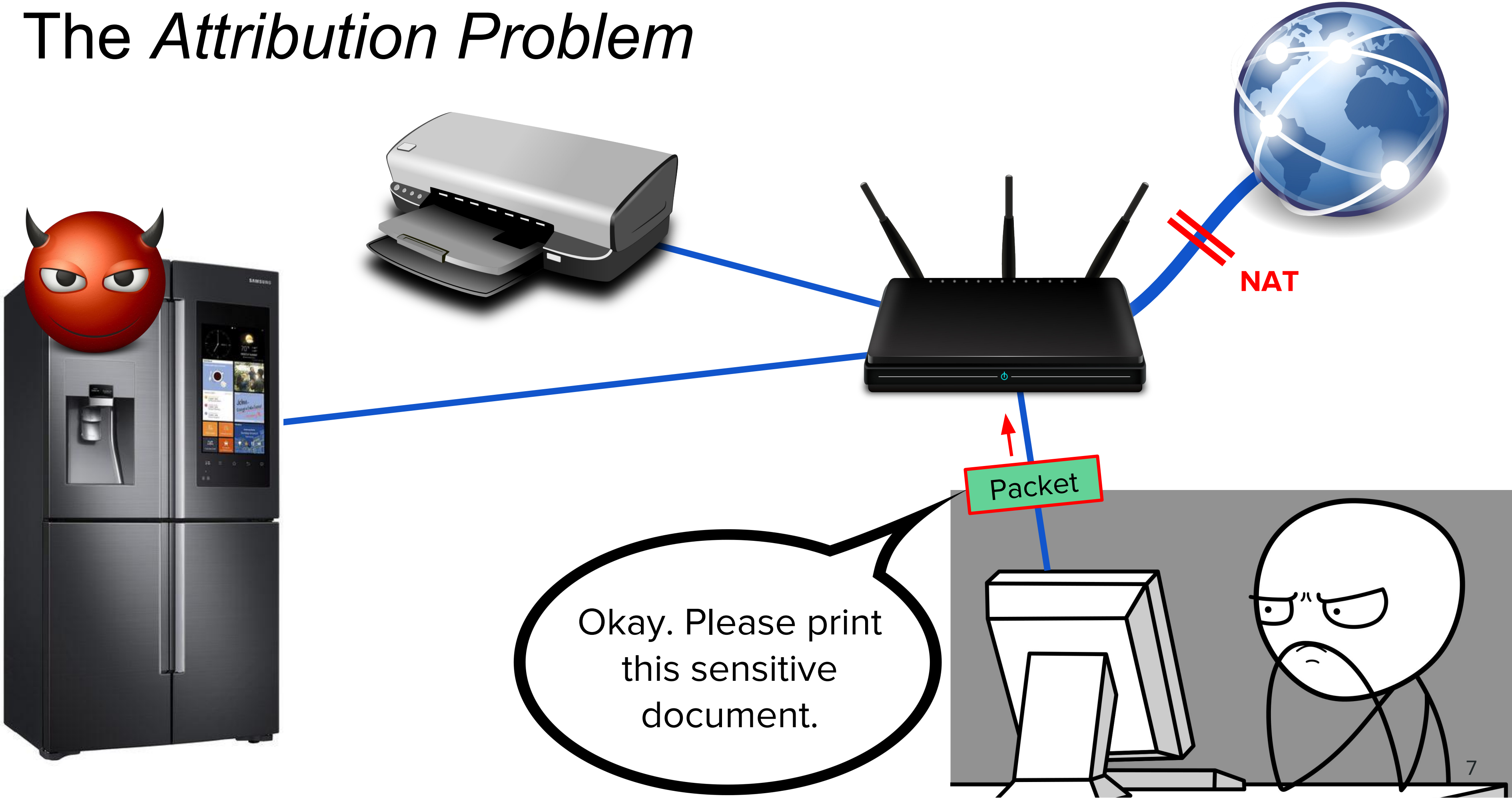


Yes. I am “Brother DCP-L2540DW series”

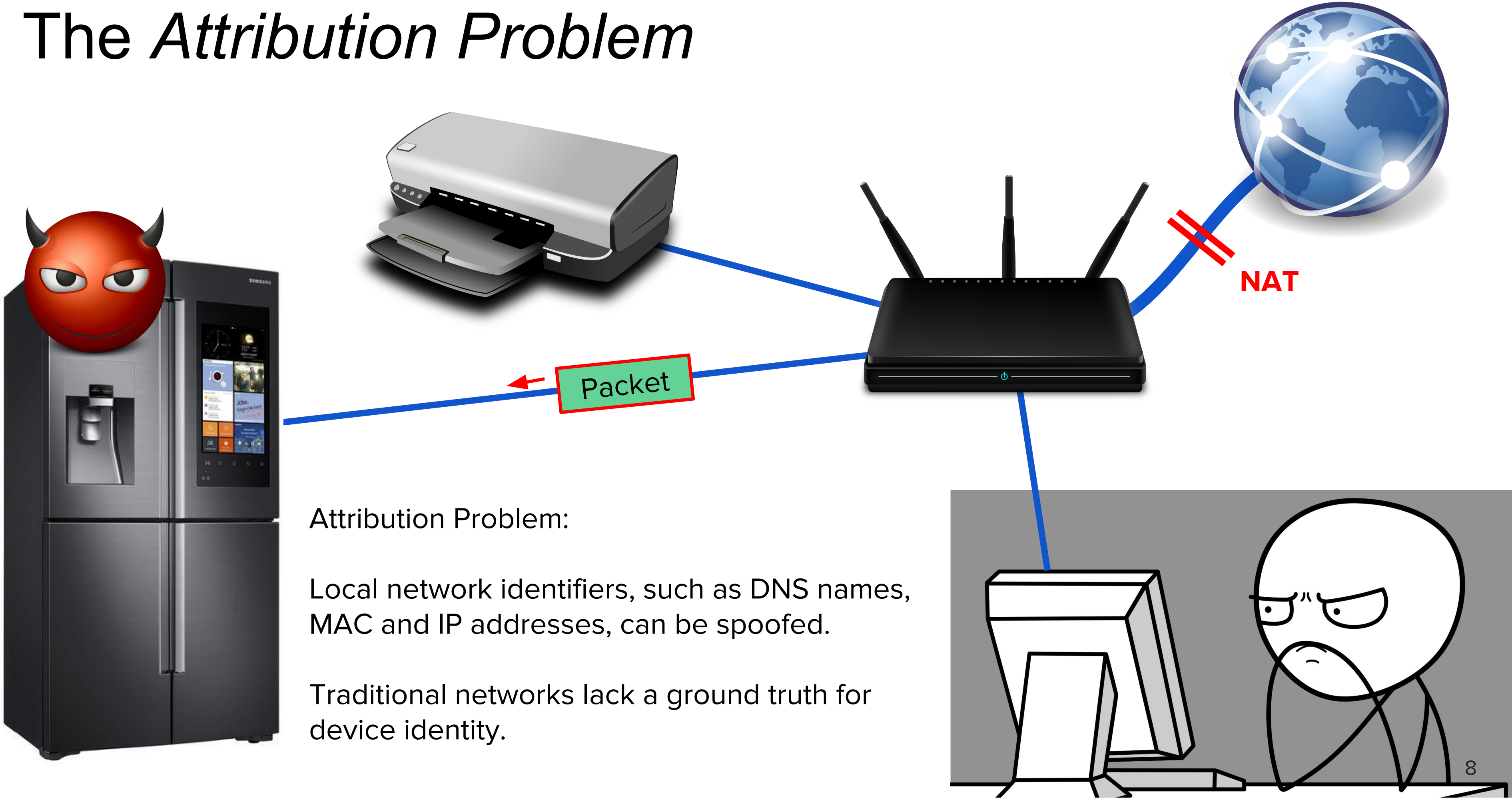
The *Attribution Problem*



The *Attribution Problem*



The *Attribution Problem*



Attribution Problem:

Local network identifiers, such as DNS names, MAC and IP addresses, can be spoofed.

Traditional networks lack a ground truth for device identity.

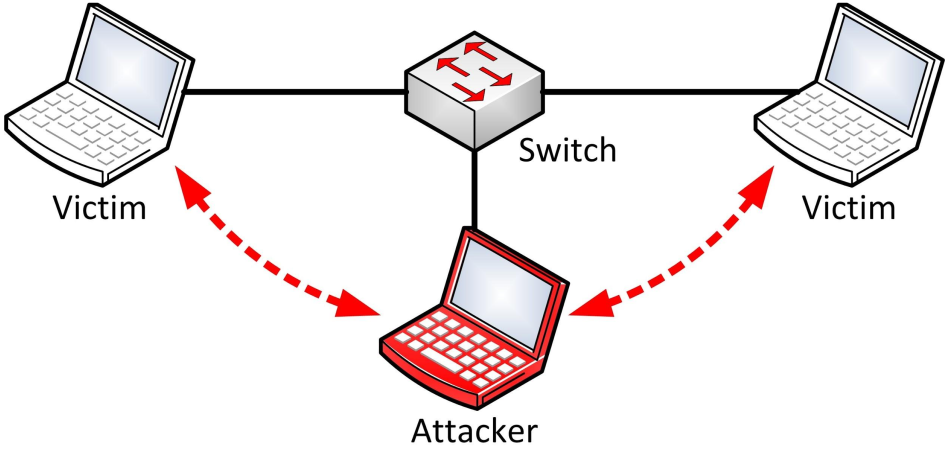
Categories of local network attacks

ARP and MAC spoofing

Name poisoning (mDNS)

Server registration spoofing

Direct attacks



The Status Quo

Intrusion Prevention in a box

IPS for the small network

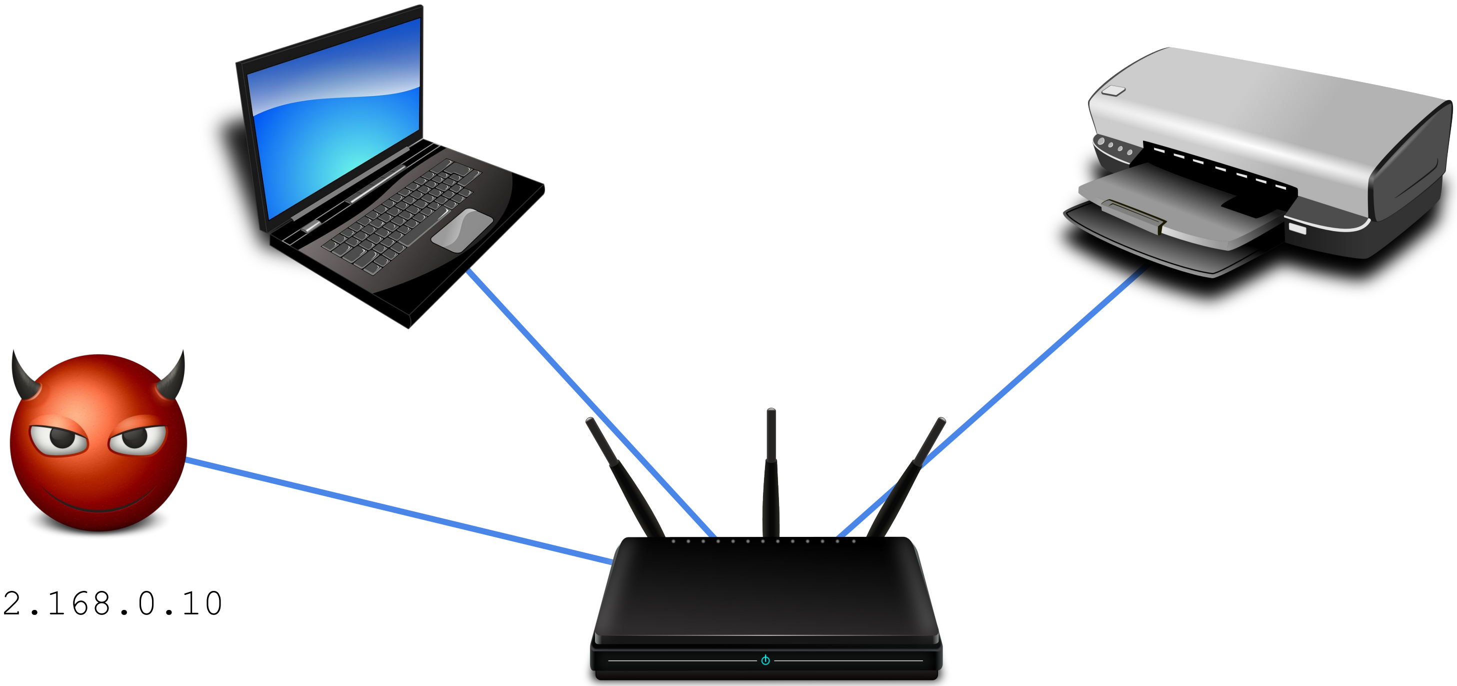
Eliminates need for local expert-level administrator

Outsources analysis to the cloud

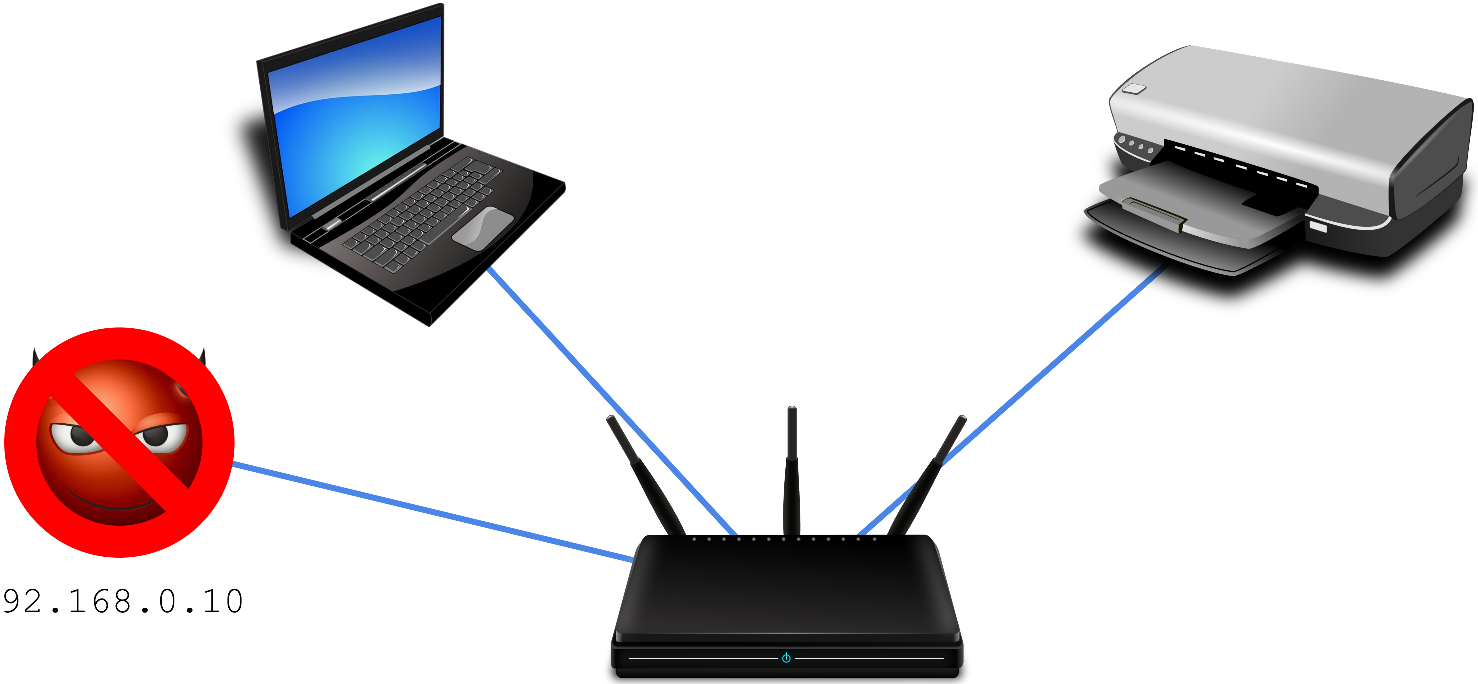
Typically more expensive and requires a subscription fee

Because attacks are always evolving, Intrusion Prevention is a cat-and-mouse game

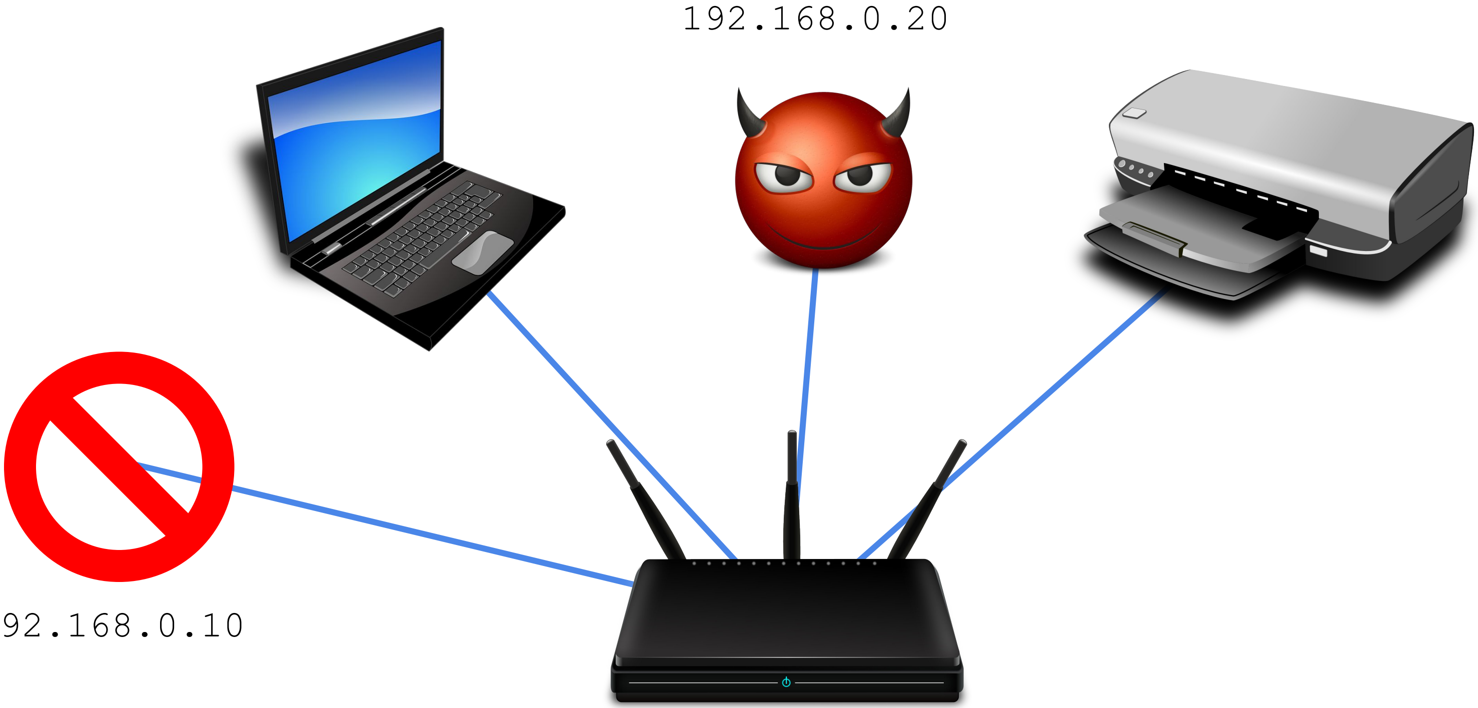
Devices can simply create and use new identifiers



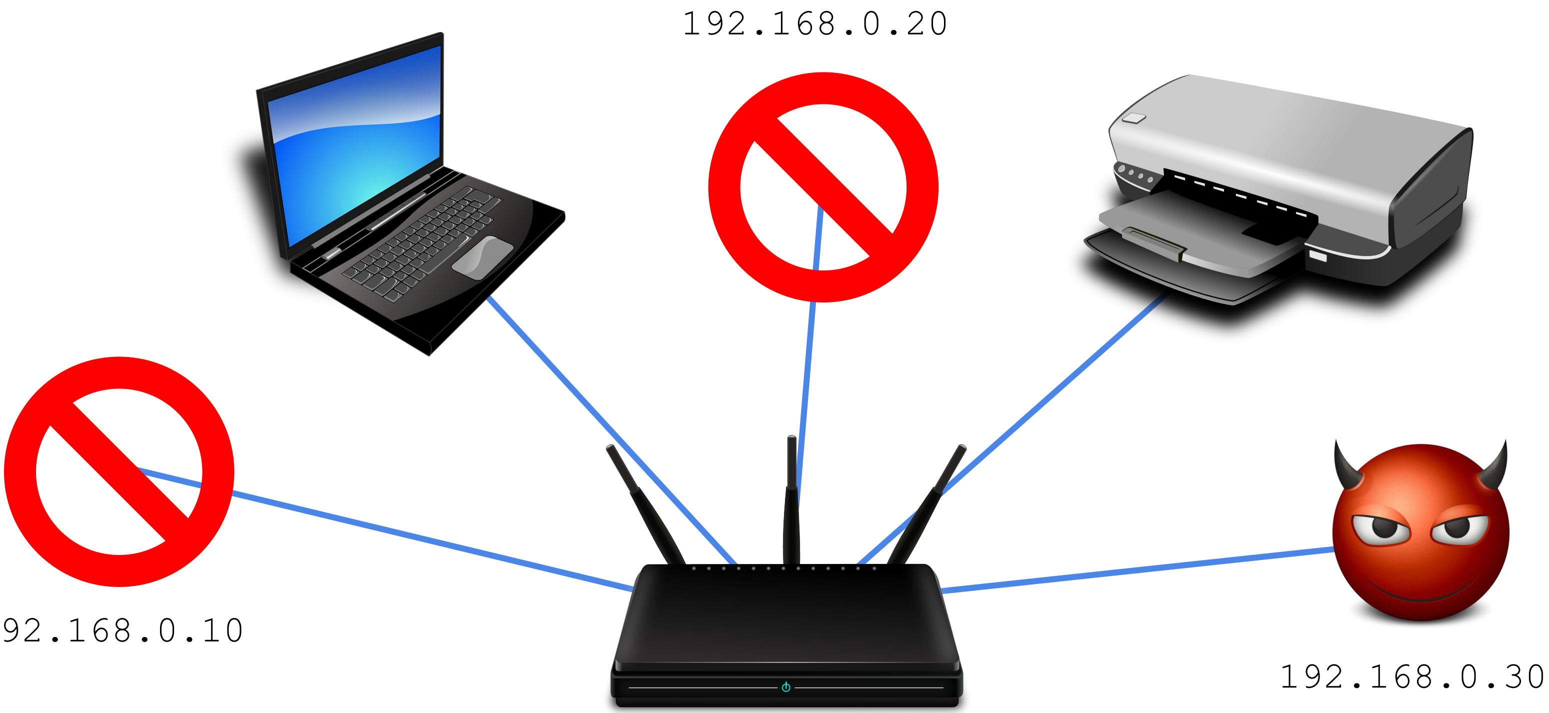
Devices can simply create and use new identifiers



Devices can simply create and use new identifiers



Devices can simply create and use new identifiers



Solutions from the literature

Soteris Demetriou et al. ***2017****. HanGuard: SDN-driven protection of smart home WiFi devices from malicious mobile apps. In 10th ACM Conference on Security and Privacy in Wireless and Mobile Networks.*

Xiaolong Bai et al. ***2016****. Staying Secure and Unprepared: Understanding and Mitigating the Security Risks of Apple ZeroConf. In IEEE Symposium on Security and Privacy.*

Seyed Kaveh Fayazbakhsh, Luis Chiang, Vyas Sekar, Minlan Yu, and Jeffrey C.Mogul. ***2014****. Enforcing Network-Wide Policies in the Presence of Dynamic Middlebox Actions using FlowTags. In 11th USENIX Symposium on Networked Systems Design and Implementation.*

Tiffany Hyun-Jin Kim et al. ***2014****. Lightweight source authentication and path validation. In ACM SIGCOMM Computer Communication Review.*

Gao Jinhua and Xia Kejian. ***2013****. ARP spoofing detection algorithm using ICMP protocol. In International Conference on Computer Communication and Informatics*

Andre Ortega, Xavier Marcos, Luis Chiang, and Cristina Abad. ***2009****. Preventing ARP Cache Poisoning Attacks: A Proof of Concept using OpenWrt. In Latin American Network Operations and Management Symposium.*

Vivek Ramachandran and Sukumar Nandi. ***2005****. Detecting ARP Spoofing: An Active Technique. In Information Systems Security. ICISS*

D. Bruschi, A. Ornaghi, and E. Rosti. ***2003****. S-ARP: a Secure Address Resolution Protocol. In Proceedings of the 19th Annual Computer Security Applications Conference. ACSAC*

M. V. Tripunitara and P. Dutta. ***1999****. A middleware approach to asynchronous and backward compatible detection and prevention of ARP cache poisoning.In 15th Annual Computer Security Applications Conference (ACSAC '99). IEEE*

Have not reached ubiquitous adoption because:

Soteris Demetriou et al. ***2017***. *[HanGuard: SDN-driven protection of smart home WiFi devices from malicious mobile apps](). In 10th ACM Conference on Security and Privacy in Wireless and Mobile Networks.*

Xiaolong Bai et al. ***2016***. *Staying Secure and Unprepared: Understanding and Mitigating the Security Risks of Apple ZeroConf. In IEEE Symposium on Security and Privacy.*

Incompatible with legacy software and standard protocols

Seyed Kaveh Fayazbakhsh, Luis Chiang, Vyas Sekar, Minlan Yu, and Jeffrey C.Mogul. ***2014***. *Enforcing Network-Wide Policies in the Presence of Dynamic Middlebox Actions using FlowTags. In 11th USENIX Symposium on Networked Systems Design and Implementation.*

Tiffany Hyun-Jin Kim et al. ***2014***. *Lightweight source authentication and path validation. In ACM SIGCOMM Computer Communication Review.*

Gao Jinhua and Xia Kejian. ***2013***. *ARP spoofing detection algorithm using ICMP protocol. In International Conference on Computer Communication and Informatics*

Do not generalize to more than one specific attack

Andre Ortega, Xavier Marcos, Luis Chiang, and Cristina Abad. ***2009***. *Preventing ARP Cache Poisoning Attacks: A Proof of Concept using OpenWrt. In Latin American Network Operations and Management Symposium.*

Vivek Ramachandran and Sukumar Nandi. ***2005***. *Detecting ARP Spoofing: An Active Technique. In Information Systems Security. ICISS*

D. Bruschi, A. Ornaghi, and E. Rosti. ***2003***. *S-ARP: a Secure Address Resolution Protocol. In Proceedings of the 19th Annual Computer Security Applications Conference. ACSAC*

M. V. Tripunitara and P. Dutta. ***1999***. *A middleware approach to asynchronous and backward compatible detection and prevention of ARP cache poisoning. In 15th Annual Computer Security Applications Conference (ACSAC '99). IEEE*

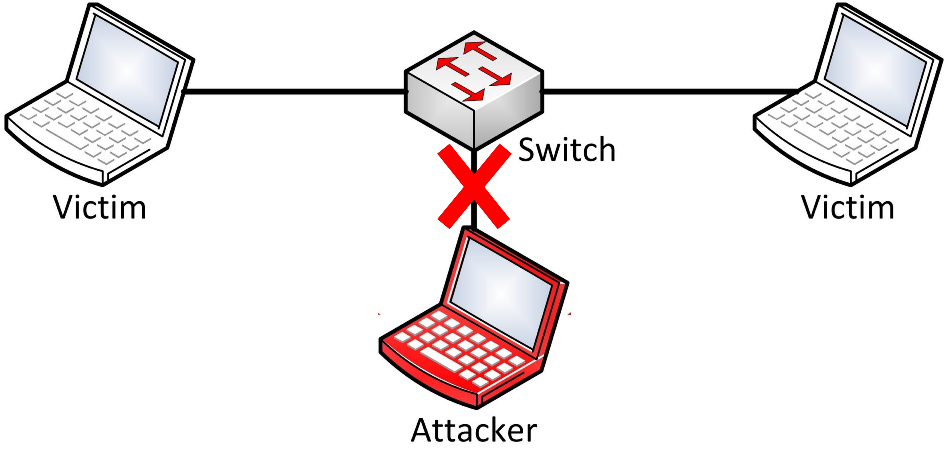
Difficult to use

Key Insight: with attribution, defense would be easy

With attribution:

Devices cannot easily masquerade as others

Blacklisted devices cannot spoof new identifiers



This enables standard access control techniques!

Attribution

How can we strongly attribute packets to devices,

without breaking *compatibility* with existing protocols?

Approach: device attribution on the central router

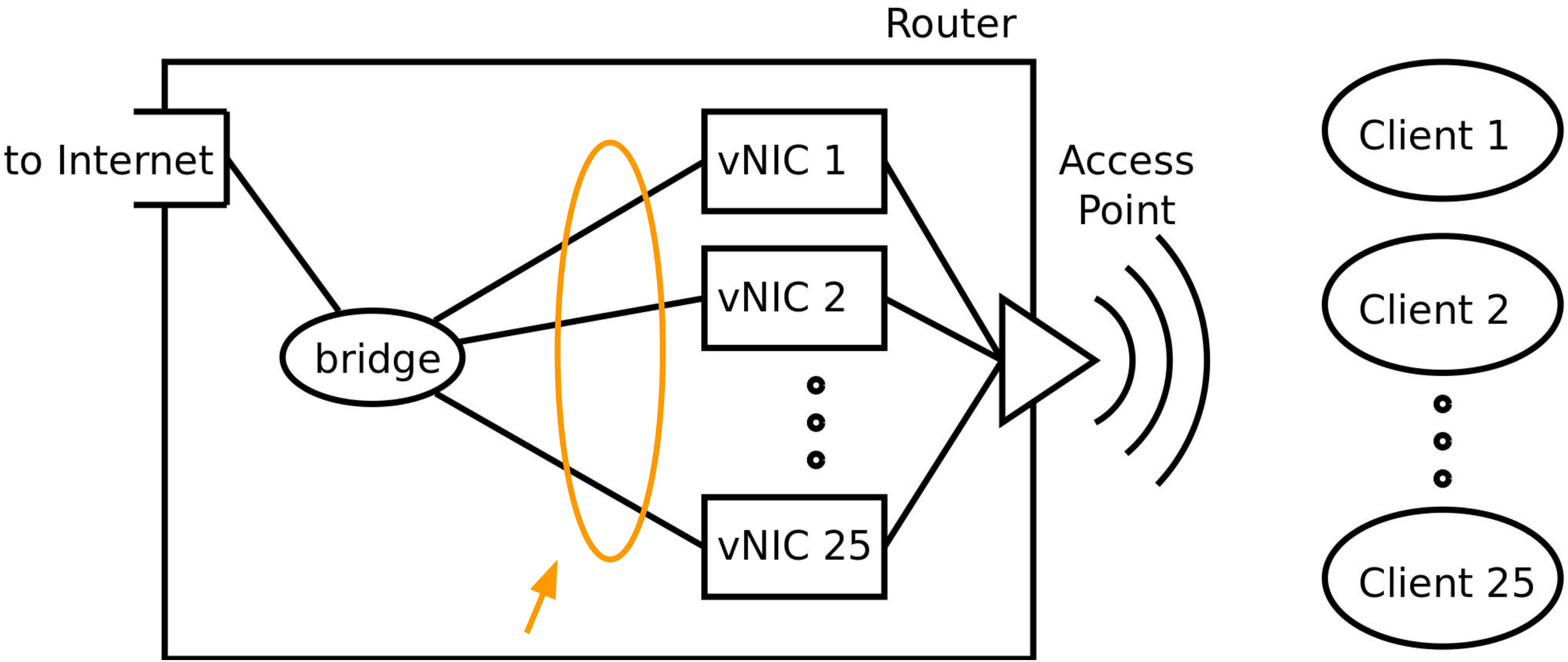
Built on the OpenWRT router OS

Supported on hundreds of consumer routers

Our prototype runs on a $50 consumer router

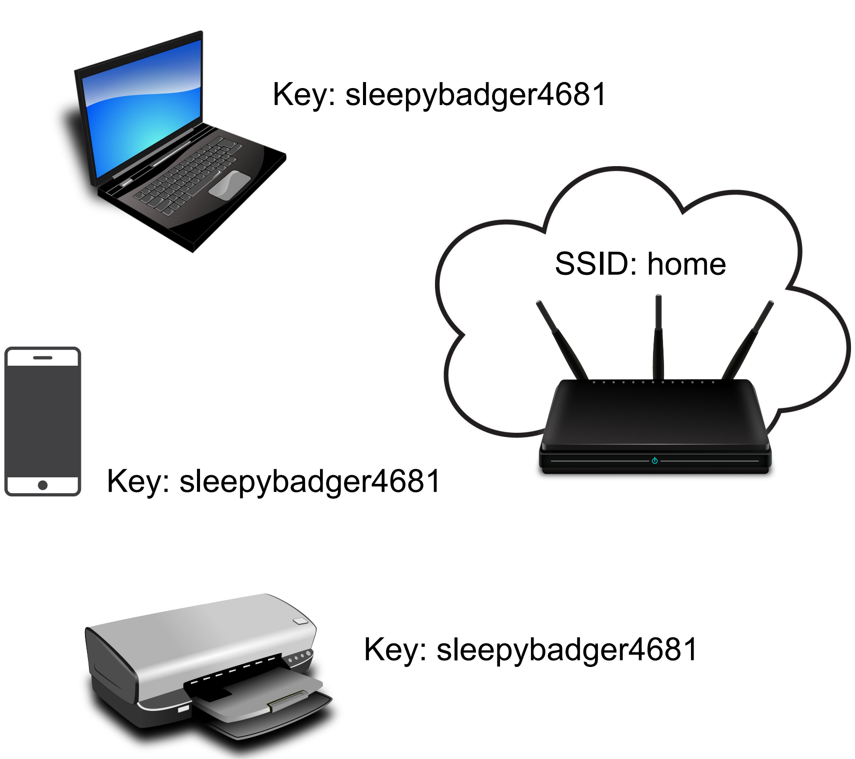
Centralized defense: no changes necessary to client devices

Goal: associate device credentials with physical layer



Can differentiate between devices by
which interface their traffic arrives on.

WPA Personal: a new model

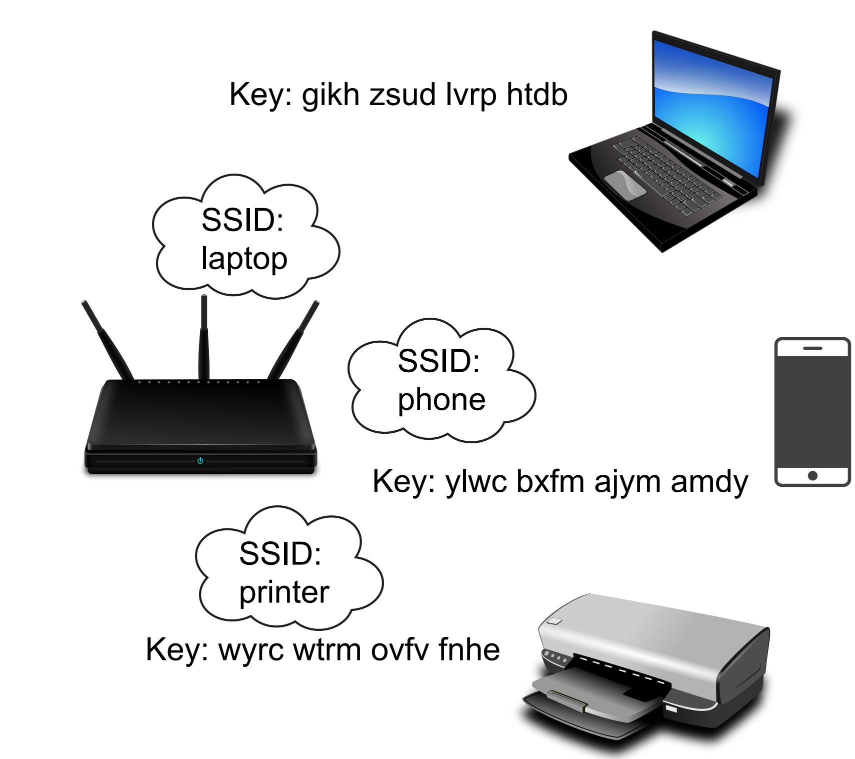


Key: sleepybadger4681

SSID: home

Key: sleepybadger4681

Key: sleepybadger4681



Key: gikh zsud lvrp htdb

SSID: laptop

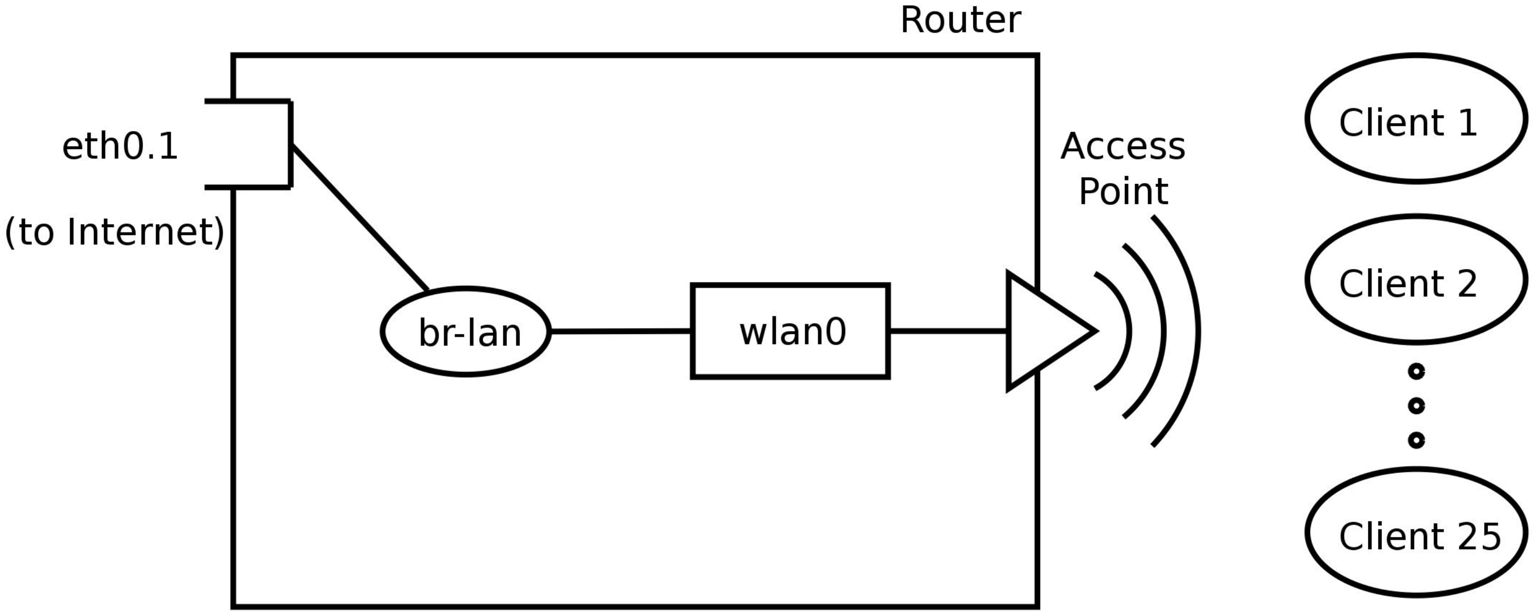
SSID: phone

Key: ylwc bxfm ajym amdy

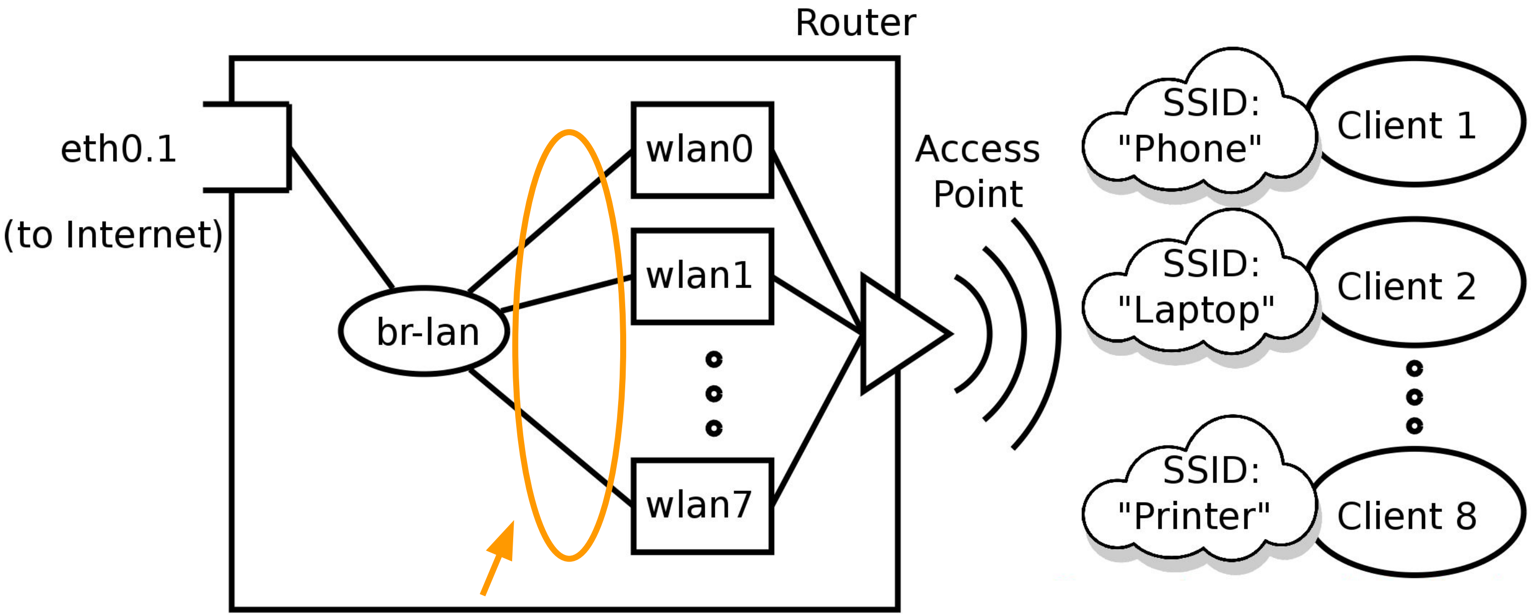
SSID: printer

Key: wyrc wtrm ovfv fnhe

Architecture - WPA Personal with shared key

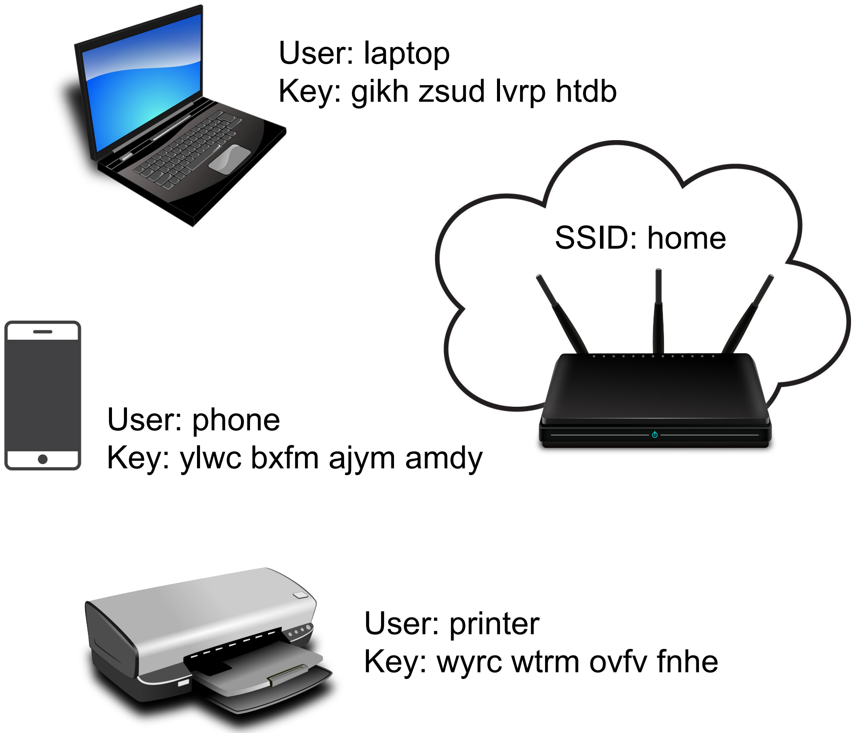


Architecture - WPA Personal with multiple SSID



Can attribute traffic to a specific client
using virtual network interface

WPA Enterprise: binding credentials to interfaces



User: laptop
Key: gikh zsud lvrp htdb

SSID: home

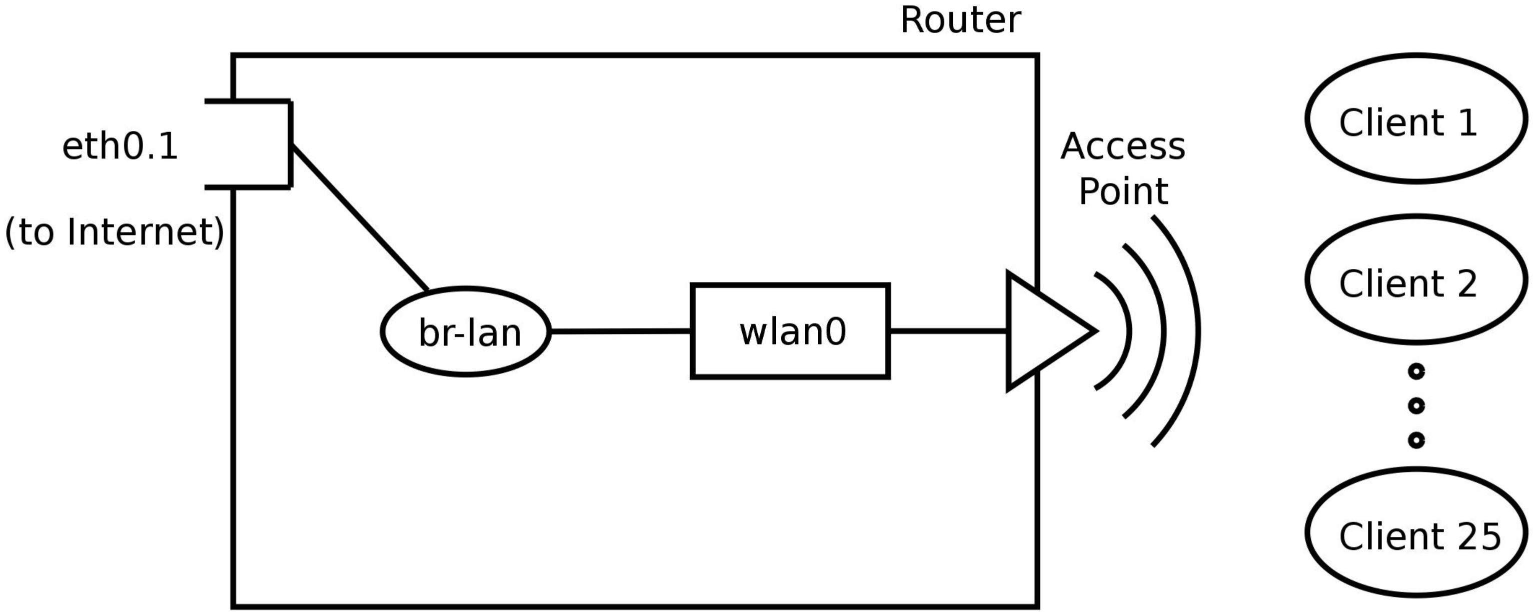
User: phone
Key: ylwc bxfm ajym amdy

User: printer
Key: wyrc wtrm ovfv fnhe

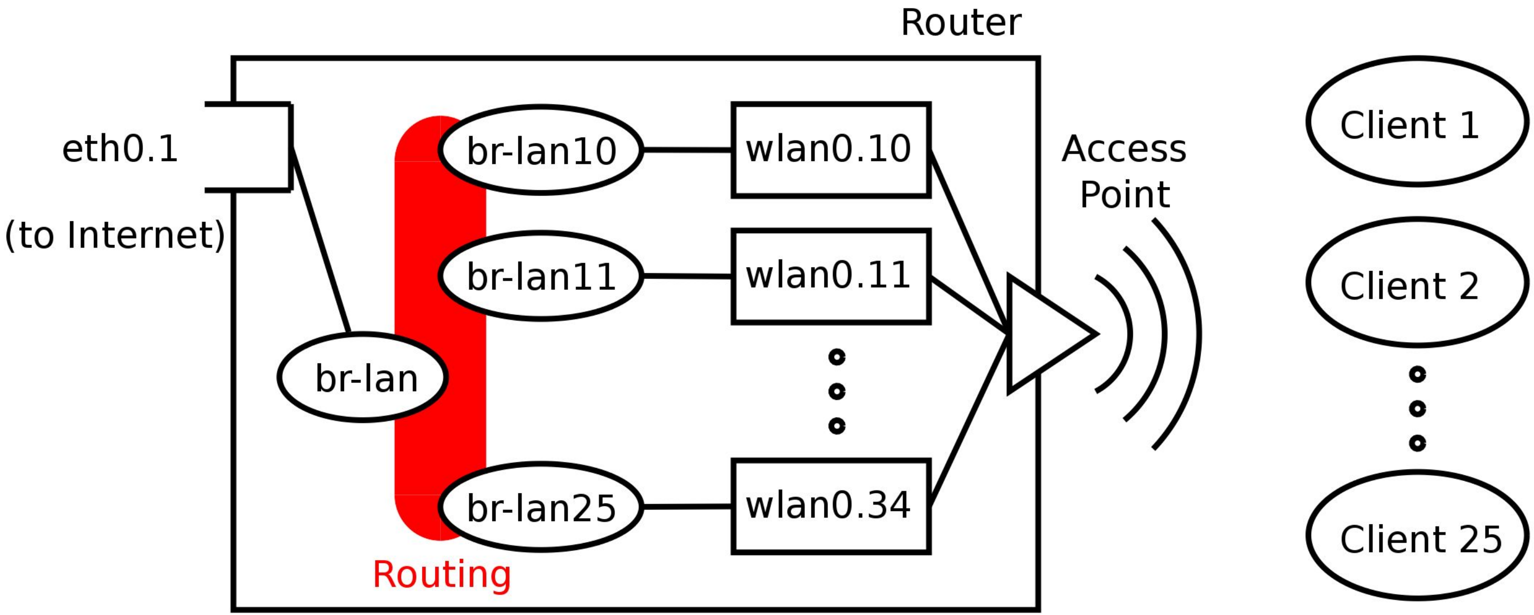
Problem: How do we put wireless clients, on the *same* wireless network, on *different* network interfaces so we can differentiate between them?

Wireless clients can be segregated into completely separate logical networks with *VLAN Isolation*

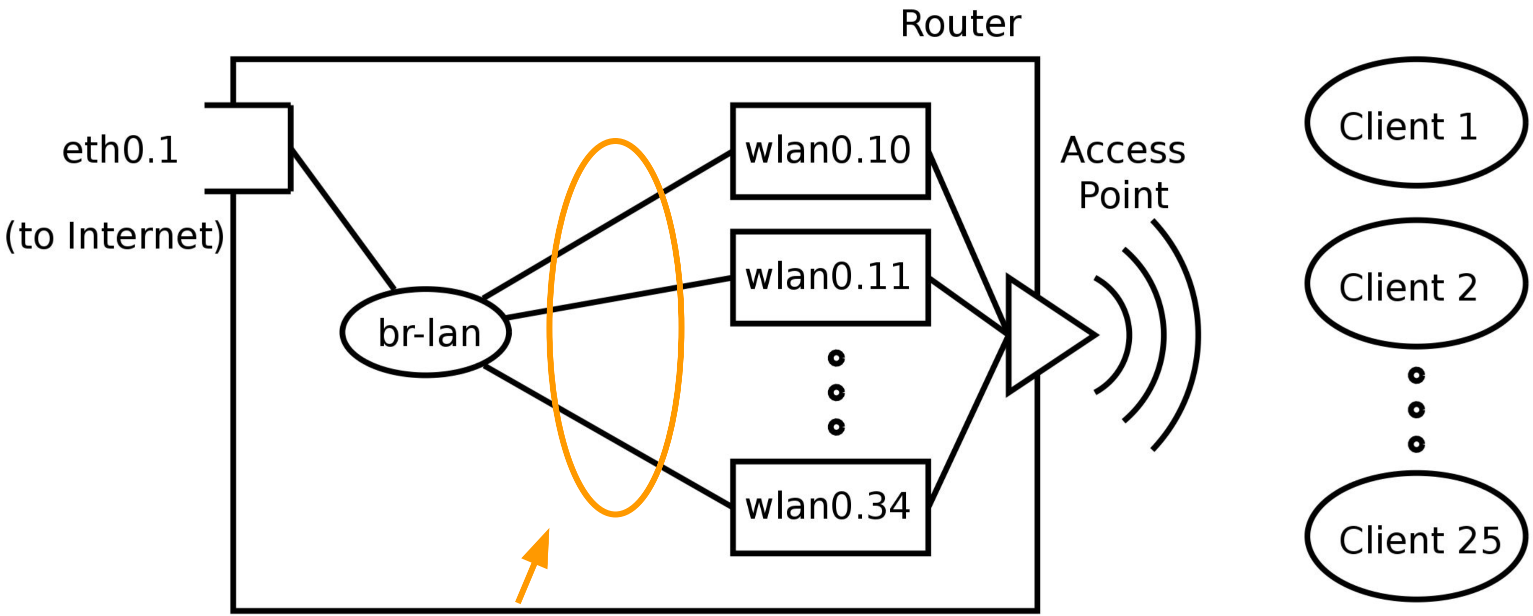
Architecture - WPA Enterprise (traditional)



Architecture - WPA Enterprise with VLAN Isolation



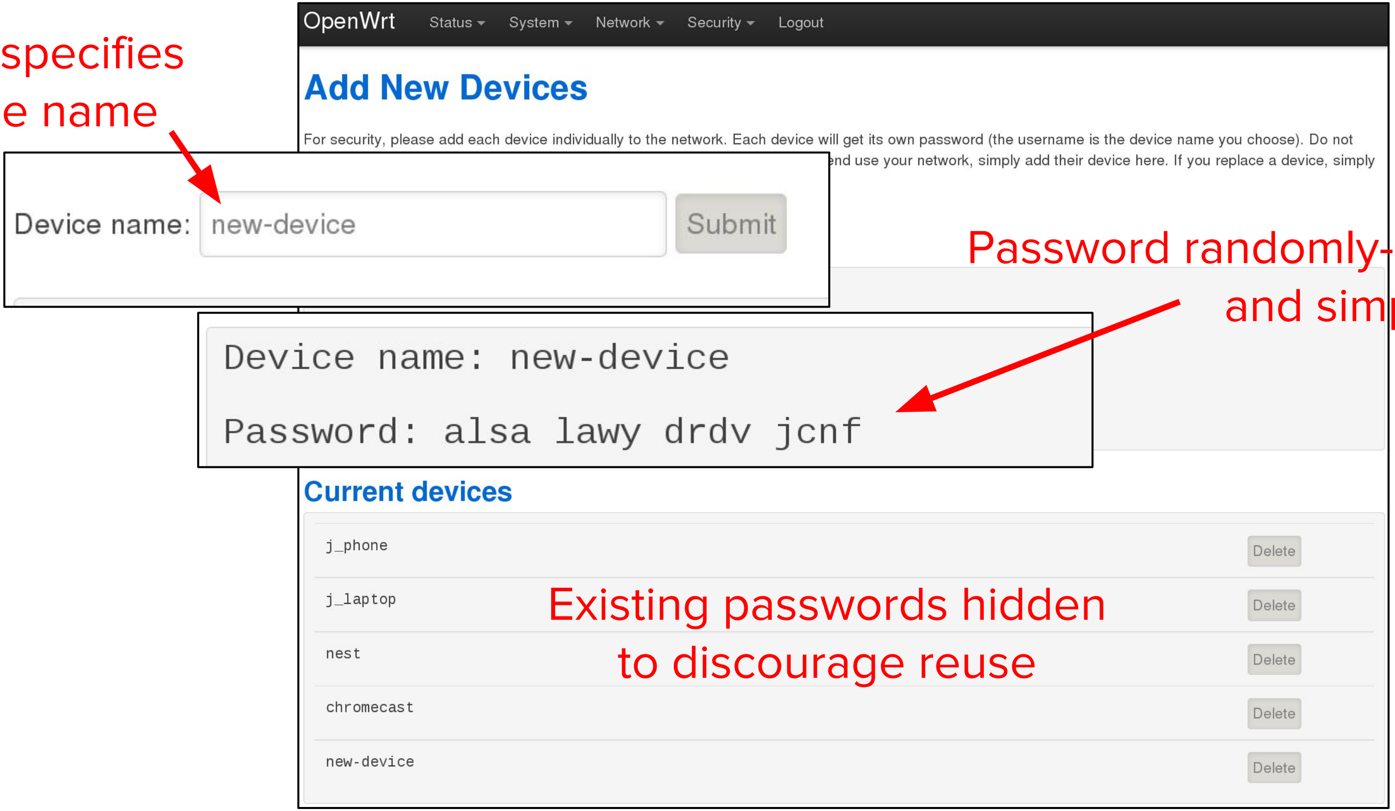
Architecture - WPA Enterprise with attribution



Bridge networks together, enabling local traffic
to propagate while retaining attribution.

Automated configuration: easy to add new devices

User specifies device name



Add New Devices

For security, please add each device individually to the network. Each device will get its own password (the username is the device name you choose). Do not

end use your network, simply add their device here. If you replace a device, simply

Device name: new-device

```
Device name: new-device

Password: alsa lawy drdv jcnf
```

Password randomly-generated and simple to type

Current devices

j_phone	Delete
j_laptop	Delete
nest	Delete
chromecast	Delete
new-device	Delete

Existing passwords hidden to discourage reuse

Building on attribution - Two simple security modules

Dreamcatcher	Name poisoning Server registration spoofing Direct attacks
Checkpoint	ARP spoofing MAC spoofing

You can certainly build additional modules to achieve additional goals

Dreamcatcher

Demand-driven, user-informed access control

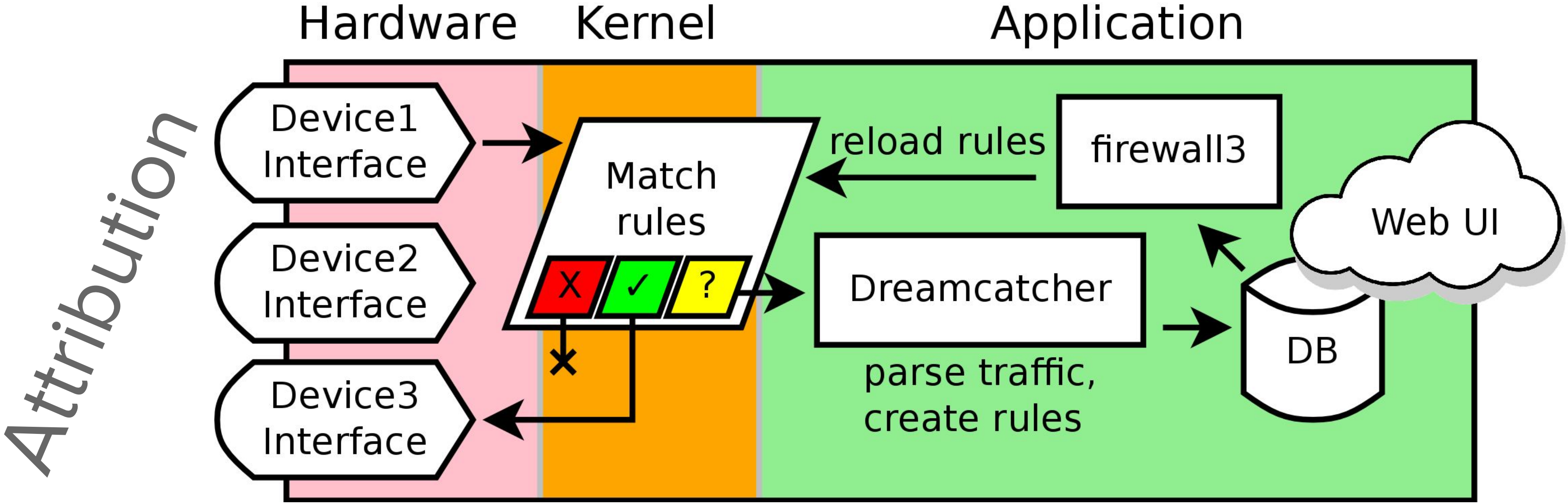
Defends against:

Name poisoning

Server registration spoofing

Direct attacks

Overview of Dreamcatcher

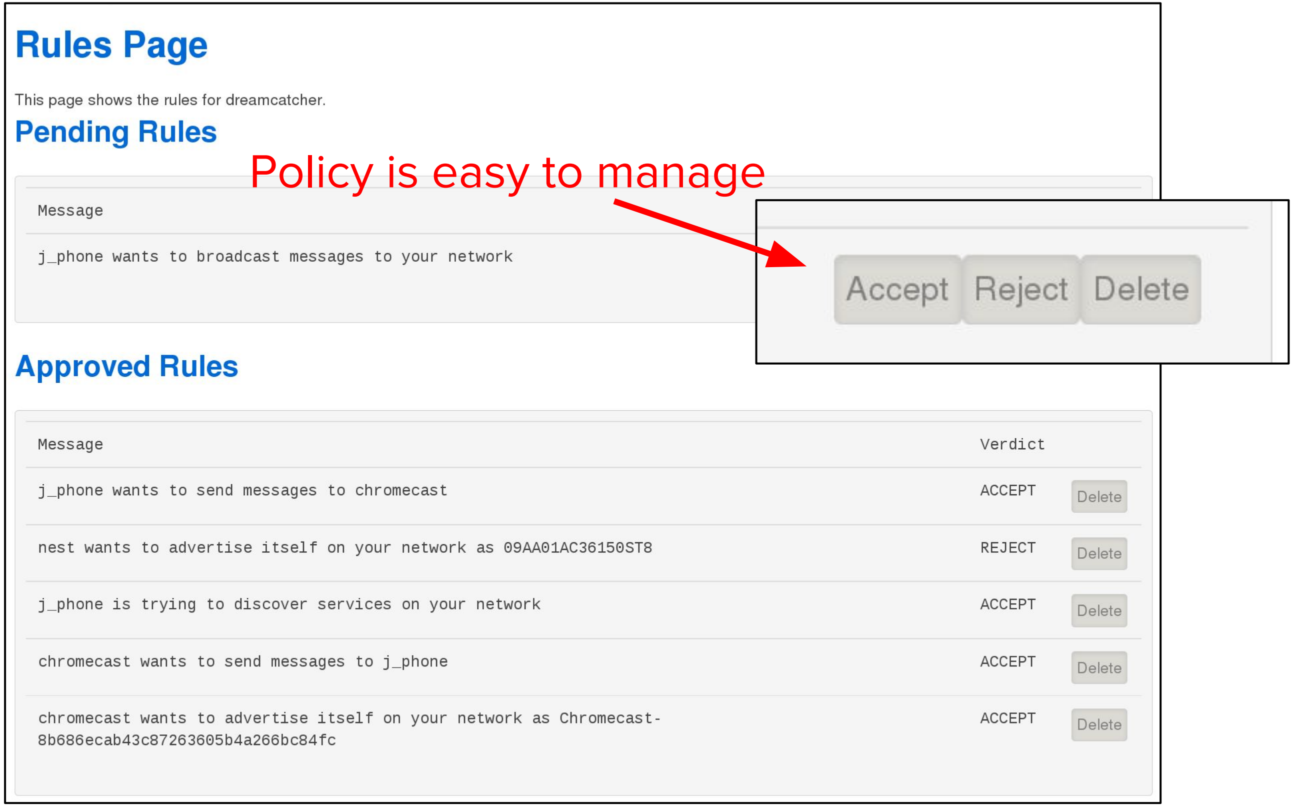


Key insight: users have contextual awareness of desired network function

Leverage user context to form access control policy

User-informed access control policy

Instant feedback! Companion app alerts user whenever a new rule is created



Rules Page

This page shows the rules for dreamcatcher.

Pending Rules

Message
j_phone wants to broadcast messages to your network

Policy is easy to manage

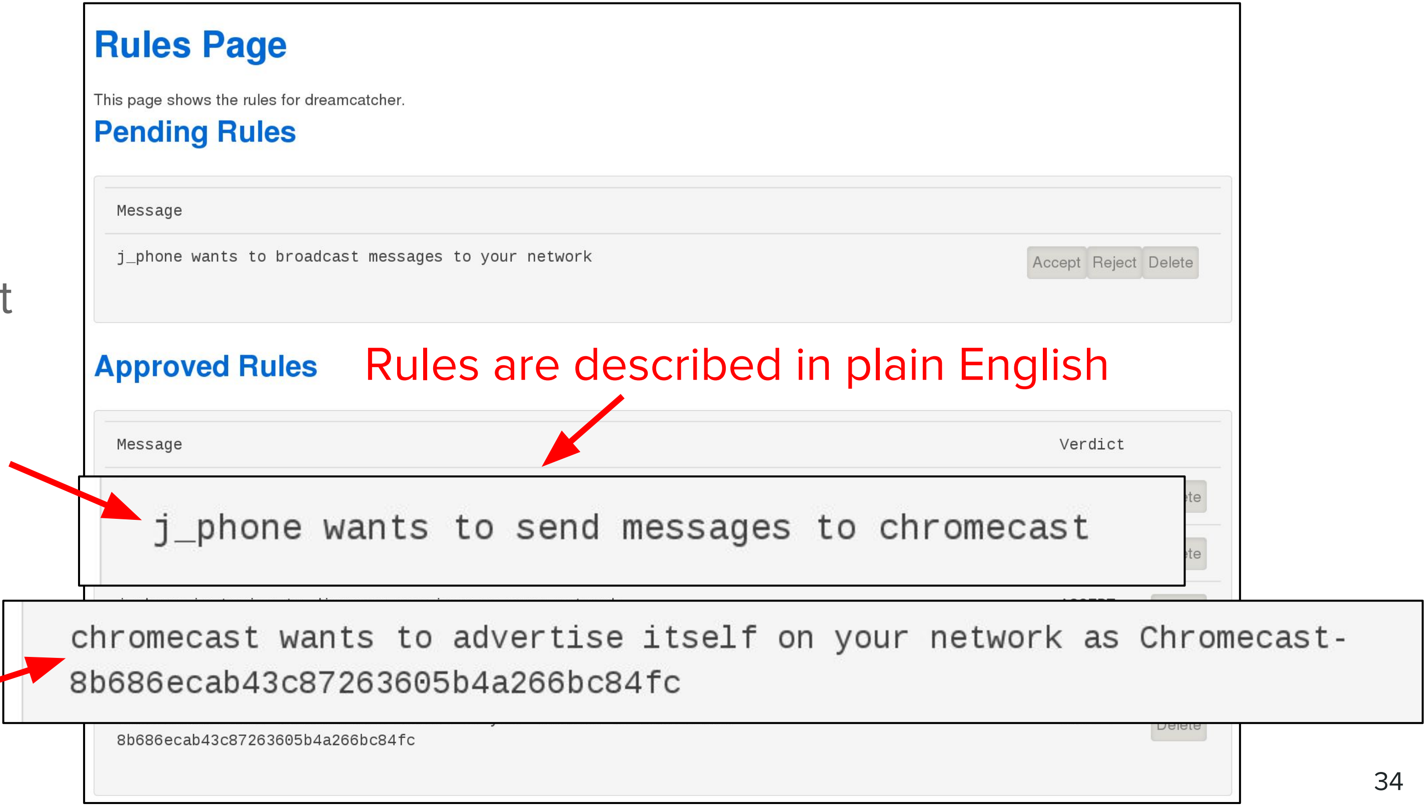
Approved Rules

Message	Verdict	
j_phone wants to send messages to chromecast	ACCEPT	Delete
nest wants to advertise itself on your network as 09AA01AC36150ST8	REJECT	Delete
j_phone is trying to discover services on your network	ACCEPT	Delete
chromecast wants to send messages to j_phone	ACCEPT	Delete
chromecast wants to advertise itself on your network as Chromecast-8b686ecab43c87263605b4a266bc84fc	ACCEPT	Delete

User-informed access control policy

Four rule types:

- Direct
- Broadcast
- Discovery
- Advertisement



Rules Page

This page shows the rules for dreamcatcher.

Pending Rules

```
Message

j_phone wants to broadcast messages to your network
```

Approved Rules

```
Message                                                        Verdict

j_phone wants to send messages to chromecast

chromecast wants to advertise itself on your network as Chromecast-
8b686ecab43c87263605b4a266bc84fc

8b686ecab43c87263605b4a266bc84fc
```

Rules are described in plain English

Direct connections

Advertisements

Protection against attacks

Name poisoning

Users can prevent devices from masquerading under false names

Server registration spoofing

Connections to attacker devices must be explicitly allowed

Direct attacks

Untrusted devices cannot initiate new connections

This may not be 100% effective, but is substantially better than a traditional network

Checkpoint

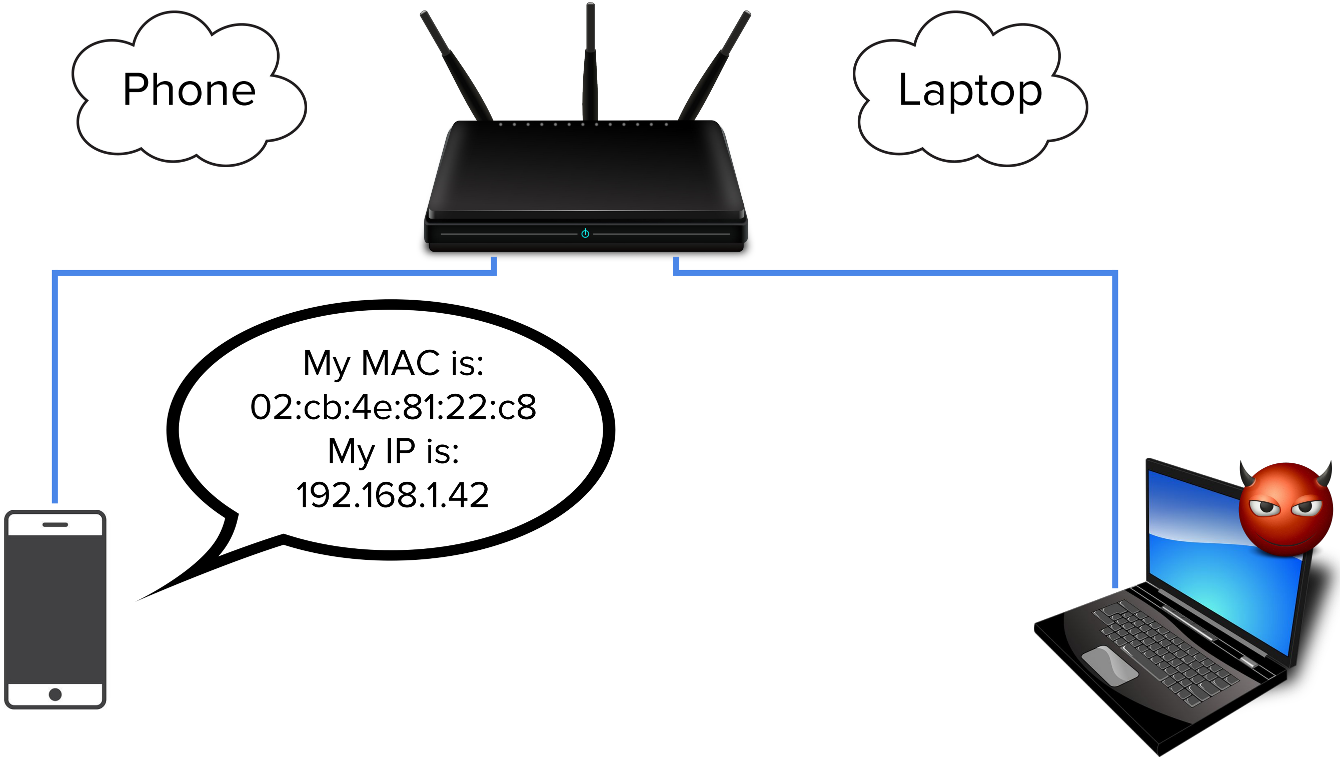
Fully-automated passive defense

Defends against:

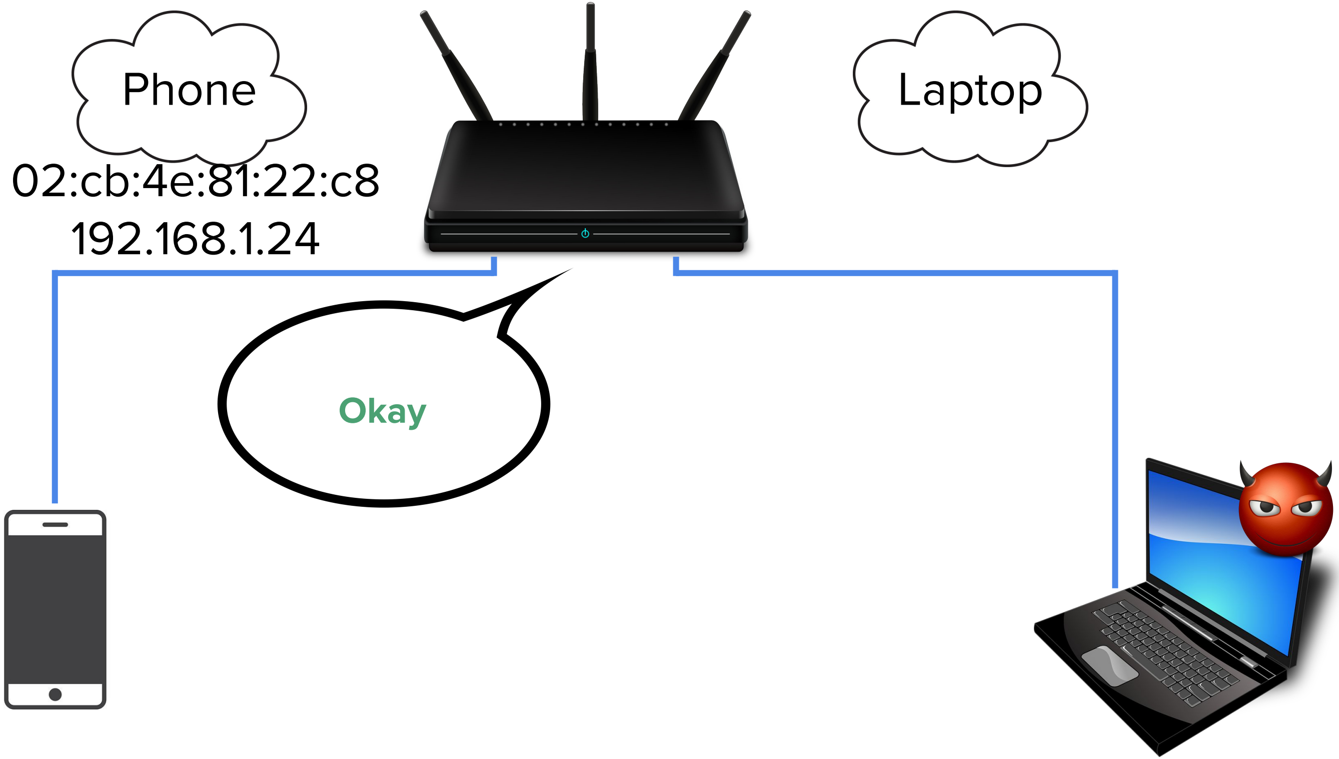
ARP spoofing

MAC spoofing

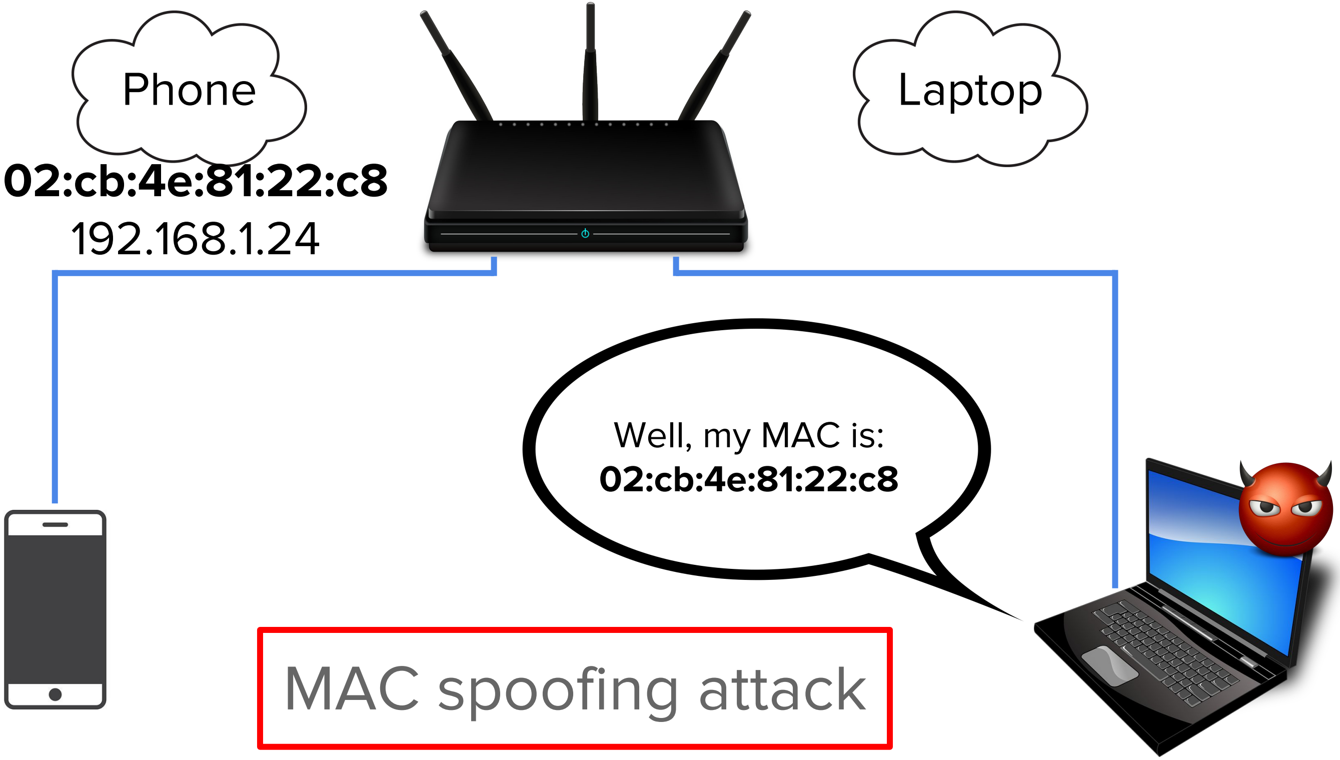
Claim-based system for Ethernet and ARP



Claim-based system for Ethernet and ARP

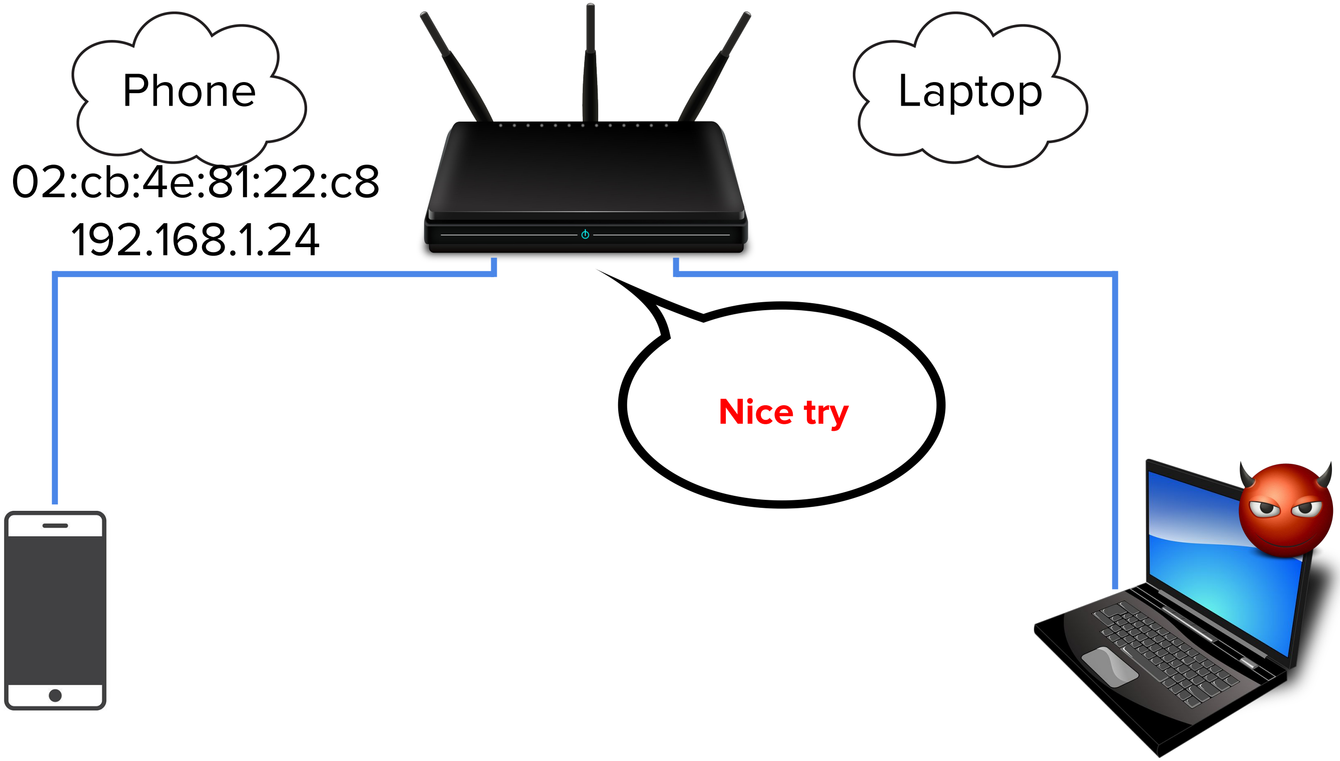


Claim-based system for Ethernet and ARP

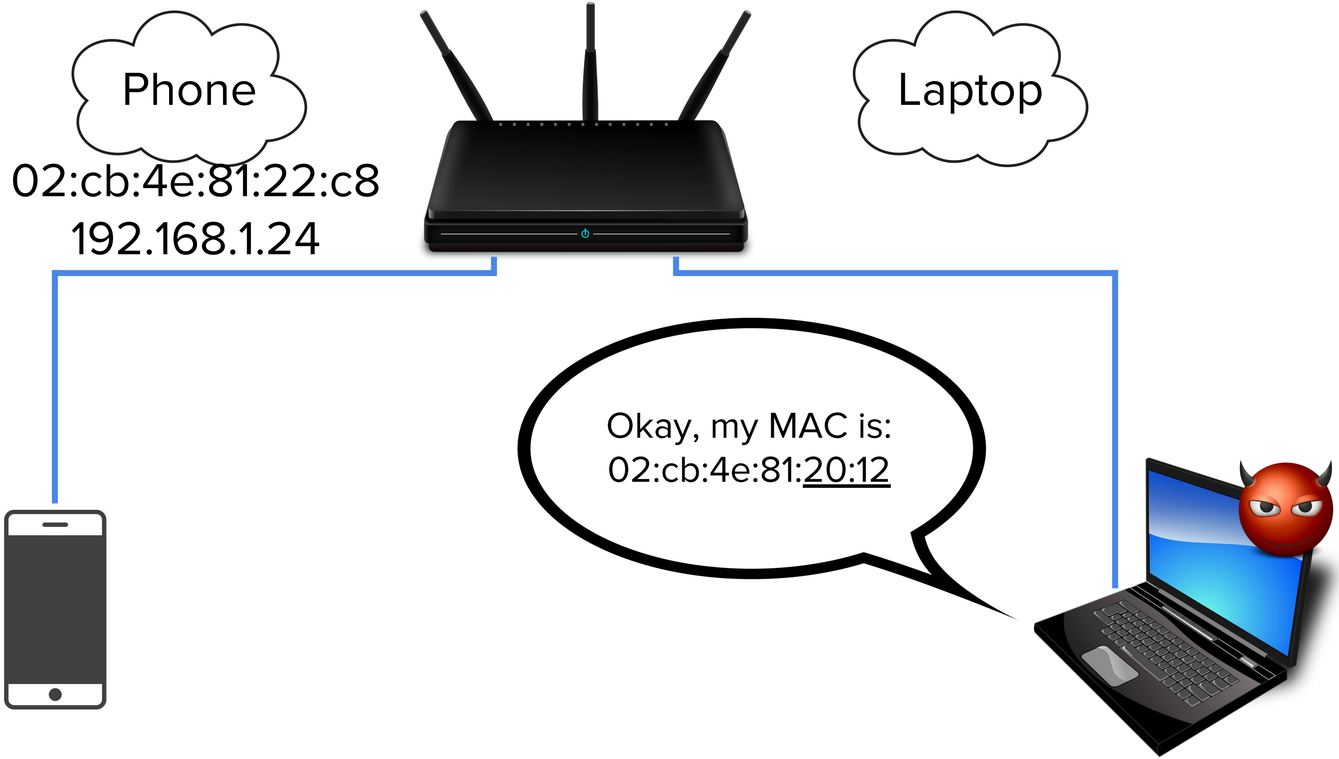


MAC spoofing attack

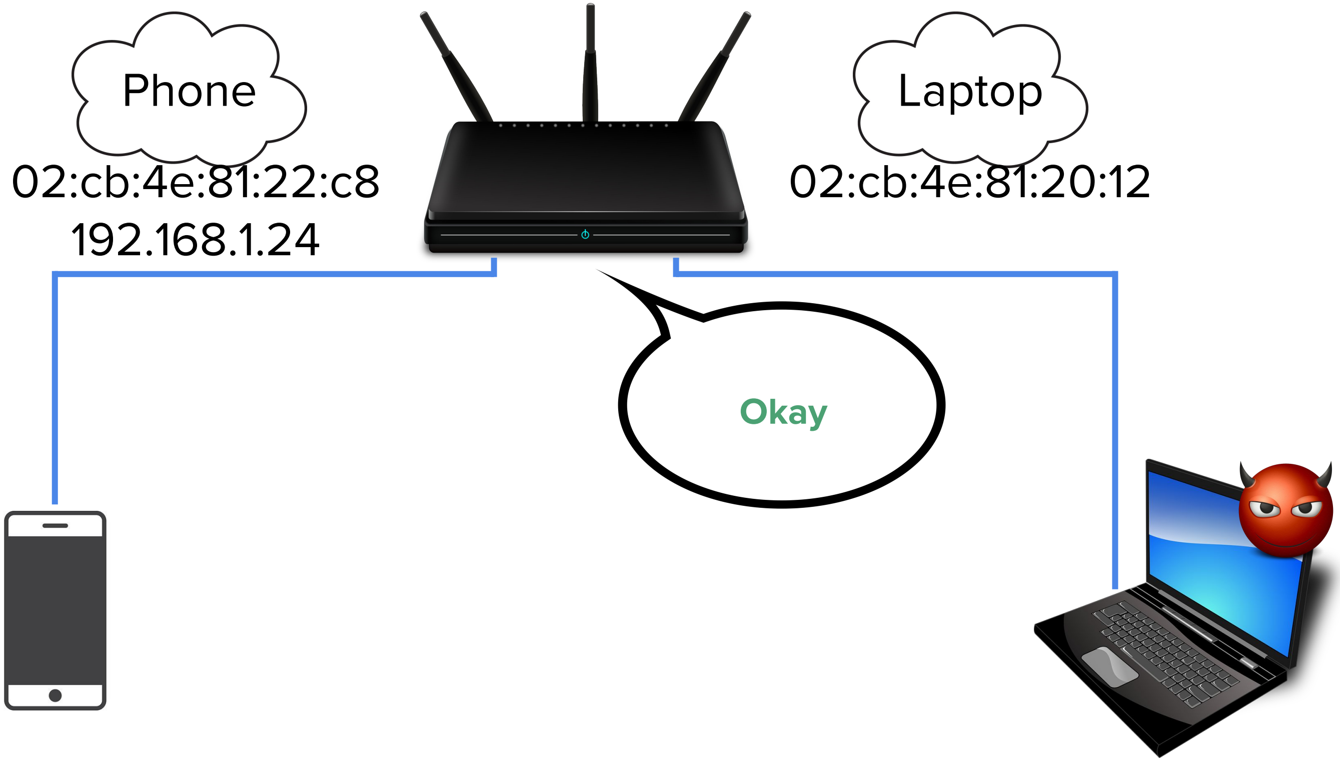
Claim-based system for Ethernet and ARP



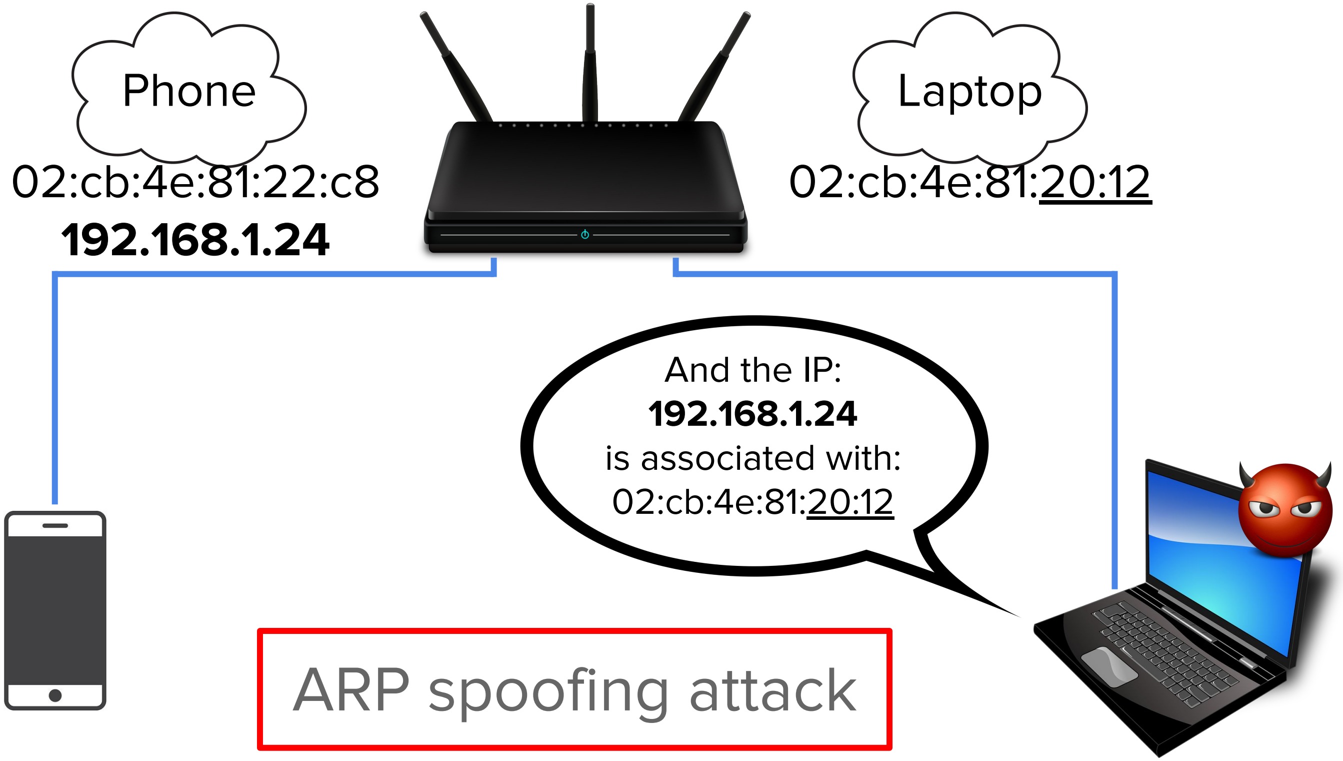
Claim-based system for Ethernet and ARP



Claim-based system for Ethernet and ARP



Claim-based system for Ethernet and ARP



Phone

02:cb:4e:81:22:c8

192.168.1.24

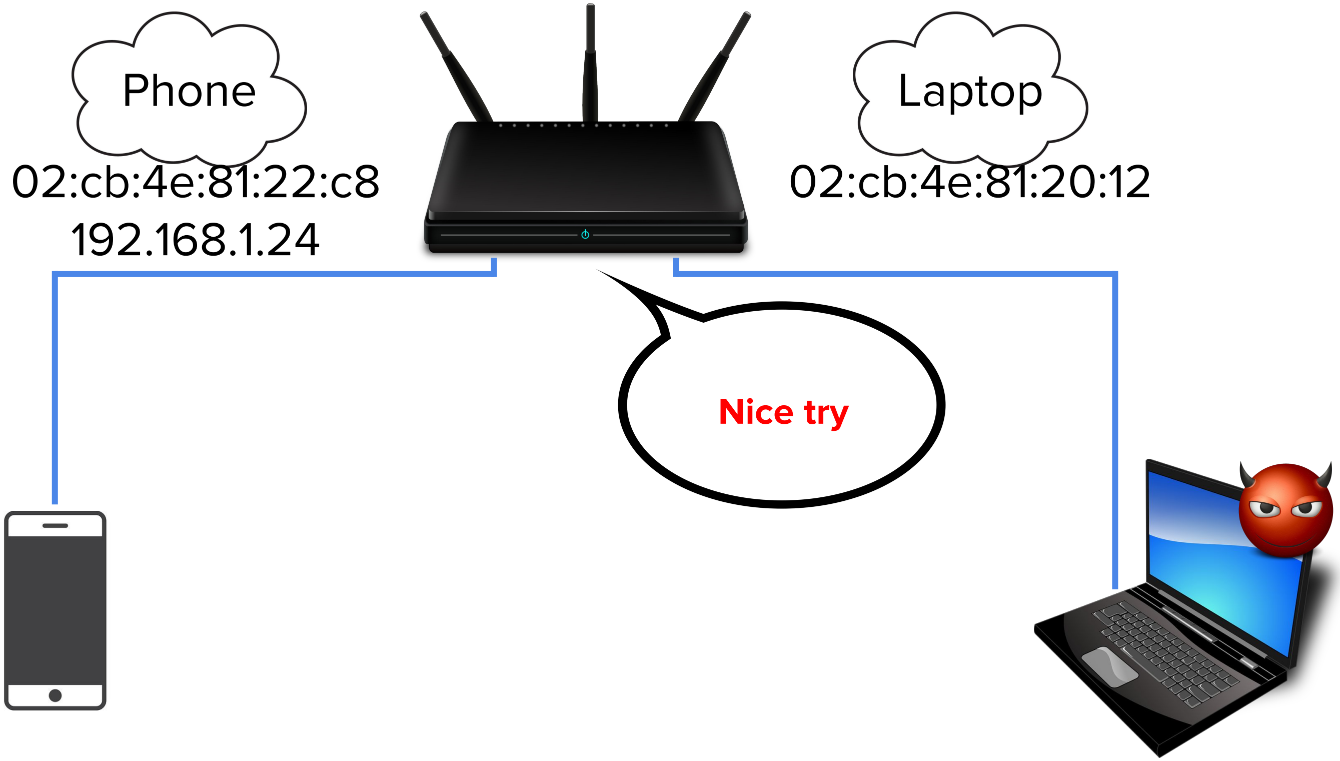
Laptop

02:cb:4e:81:20:12

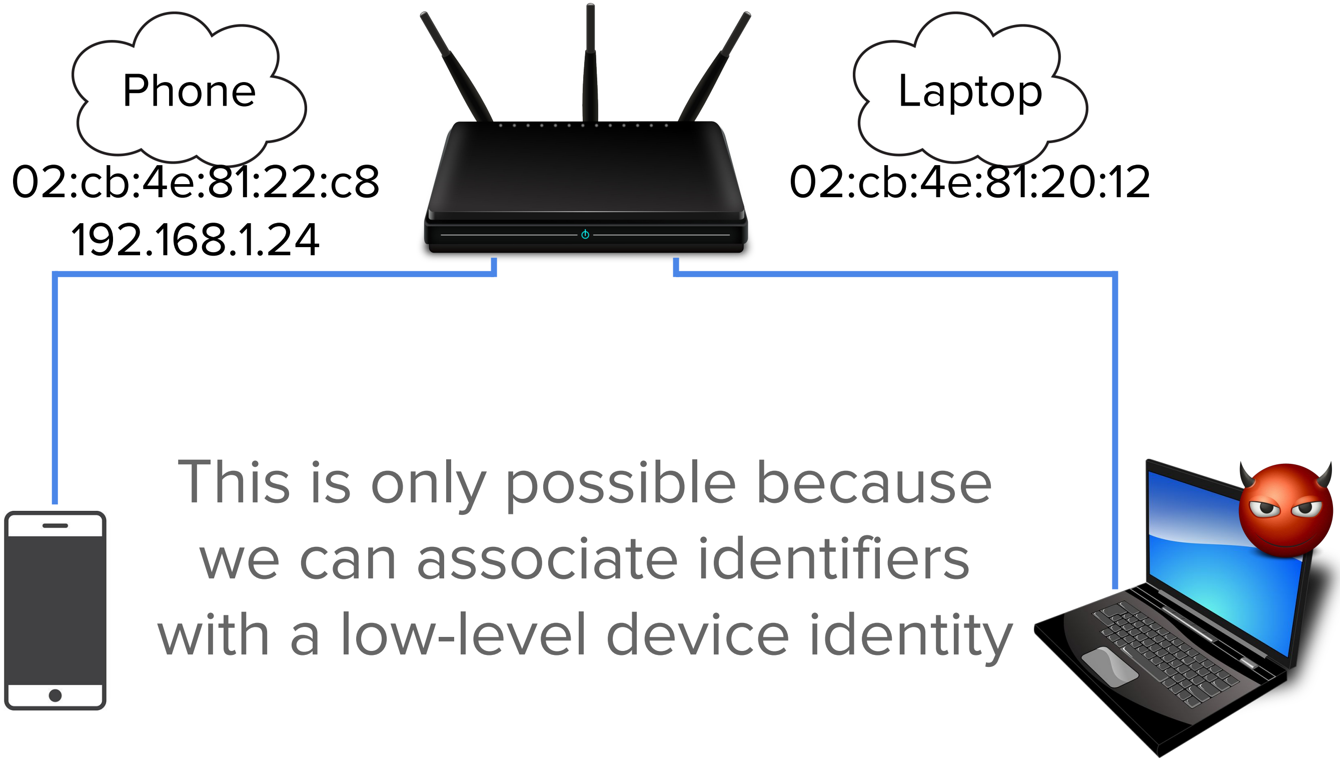
And the IP:
192.168.1.24
is associated with:
02:cb:4e:81:20:12

ARP spoofing attack

Claim-based system for Ethernet and ARP



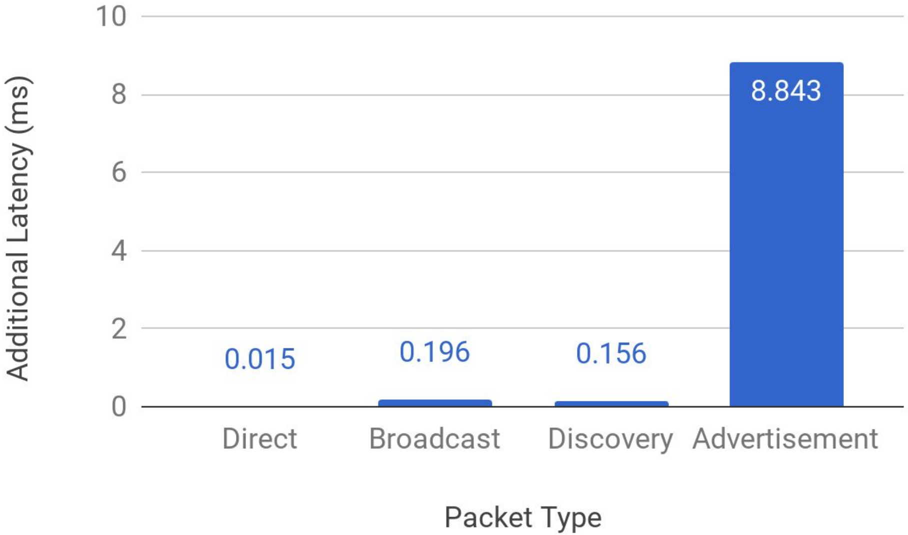
Claim-based system for Ethernet and ARP



This is only possible because we can associate identifiers with a low-level device identity

Performance and Usability

Performance: first packet latency



Packets must pass through netfilter rules from Checkpoint and Dreamcatcher.

mDNS specification, RFC 6762, mentions that mDNS responders should delay their responses by up to 500 ms.

Bandwidth change is negligible, as expected, since only the first packet traverses the rule list.

Usability study: Mechanical Turk survey

Series of scenarios following a storyline

Participants were not informed that attacks were occurring

Two setup questions

Four benign scenarios in which users must accept rules enabling devices to communicate

Three attack scenarios in which users must *not* accept rules enabling attacks to succeed

Takeaway: With limited feedback, users can make the correct rule decisions in the majority of cases.

Scenario	Success Rate
Setup	68/95 (**72%**)
Setup	82/95 (**86%**)
Benign	90/95 (**95%**)
Attack	63/95 (**66%**)
Benign	87/95 (**92%**)
Benign	74/95 (**78%**)
Benign	94/95 (**99%**)
Attack	78/95 (**82%**)
Attack	86/95 (**91%**)

Conclusion

Identification of the attribution problem

- Root cause of many small network security issues

Developed new mechanism for attributing packets to devices, which *enables*

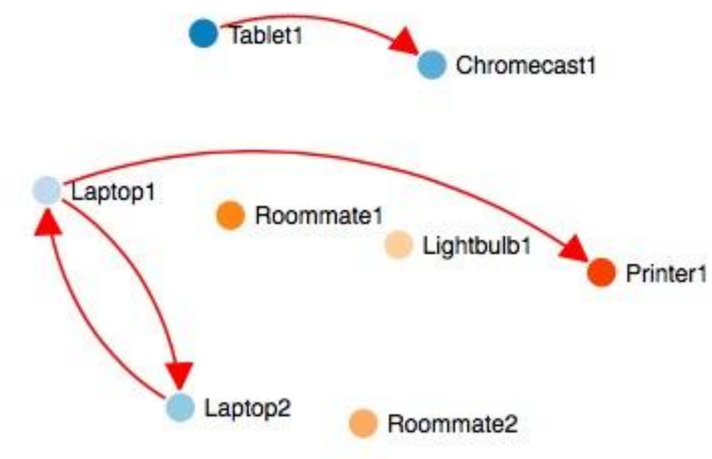
- Fully automated defense against ARP/MAC spoofing
- Strong user-informed defense against name poisoning, server registration spoofing, and direct attacks

Our solution has low overhead and is easy to use

Demo, paper, and source code can be found at
https://jeremy-erickson.com/nooneinthemiddle.html

Thank you!

Sparse network graph



Like in any access control system, privileges granted to a compromised device **can still be abused**.

However, Dreamcatcher turns a fully-connected network graph into a sparsely-connected network graph.

This limits the avenues a compromised device can use to attack new devices and allows users to set *context-aware* policies.

Combining WPA Personal and WPA Enterprise

Many *legacy* devices do not support WPA Enterprise

✓ Smartphones, tablets, laptop computers

✗ Printers, Nest thermostat, Chromecast

	Supports multiple devices	Supports legacy devices
WPA Personal	✗	✓
WPA Enterprise	✓	✗

These two techniques complement each other. By using both simultaneously, we:

- Support an **unlimited** number of *modern* devices on the primary WPA Enterprise network
- Support up to **15** *legacy* devices, each on their own WPA Personal network

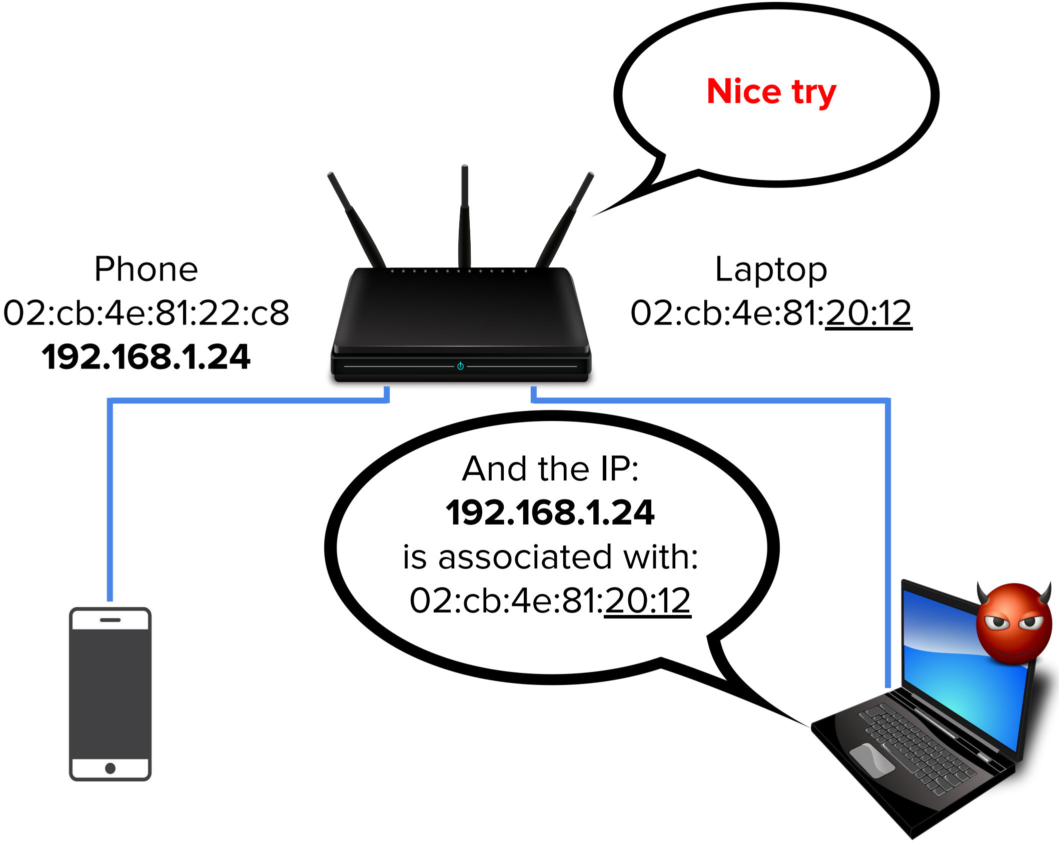
Checkpoint highlights

- Completely automated and transparent to users and devices
  - Claims are made simply by sending normal Ethernet and ARP traffic - **no protocol changes**
- Any device may claim any number of MAC and IP addresses
  - Compatible with use of bridged VMs and other non-standard use cases
  - Possible DOS attack, but not stealthy -- there are other ways to DOS the network
    - Compromised device can be easily identified and removed from network
- Claims expire if not renewed
  - Devices may leave network and be allocated a new address when they return
- Minimal performance impact
  - Filtering performed in Linux kernel (netfilter)

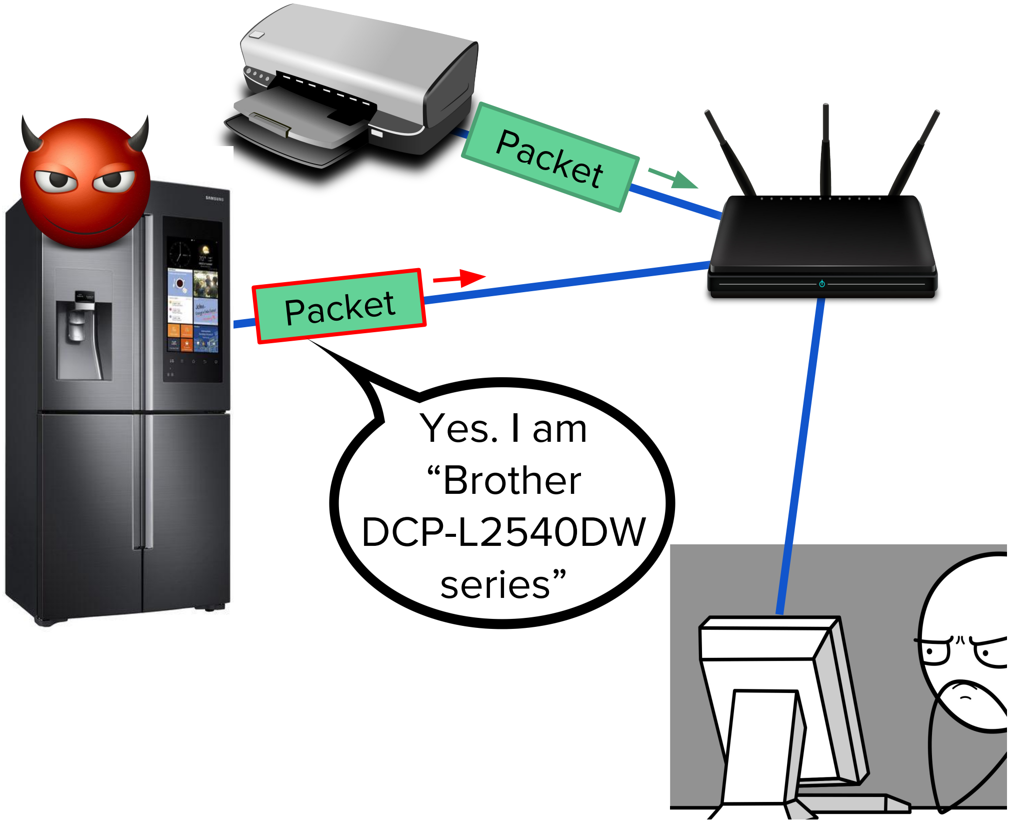
ARP spoofing

Can use Ettercap attack tool to launch ARP spoofing attack and allow Laptop to intercept communication.

With Checkpoint enabled, attack is blocked.



Name poisoning



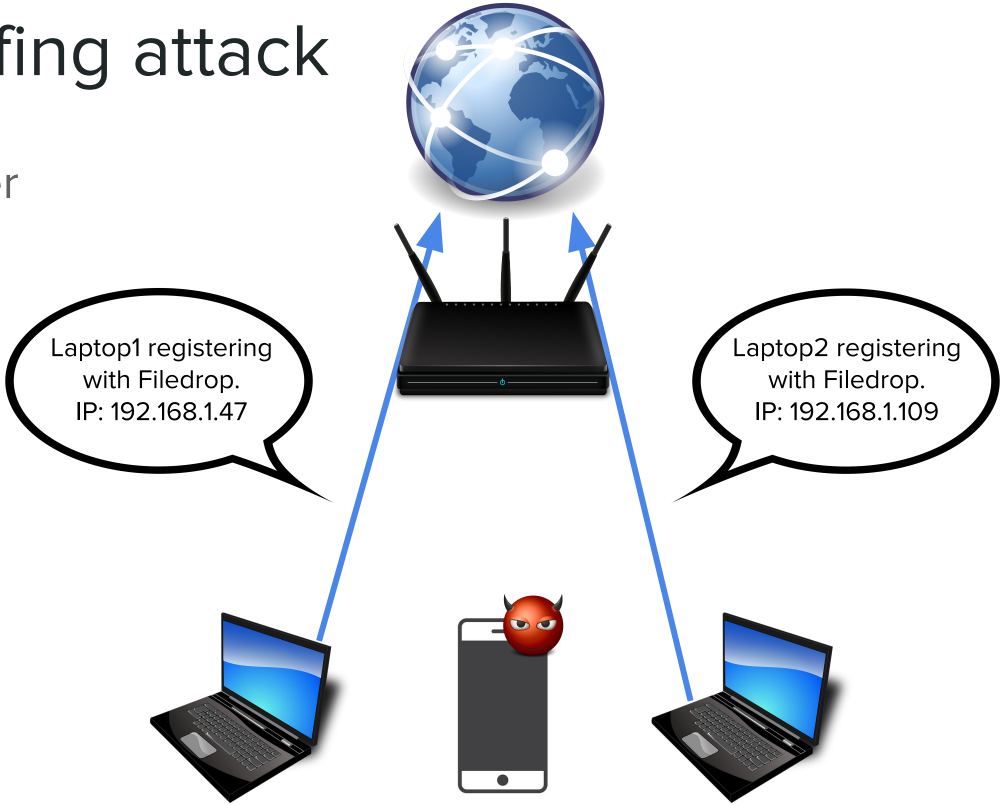
Refrigerator launches mDNS-based MitM attack to intercept printed documents between Laptop and Printer.

With Dreamcatcher, user is alerted that Refrigerator is attempting to advertise itself to the network as a printer.

By default this is blocked, and the user will most likely not allow this very suspicious advertisement.

Server registration spoofing attack

Devices use Filedrop server to register for service discovery.

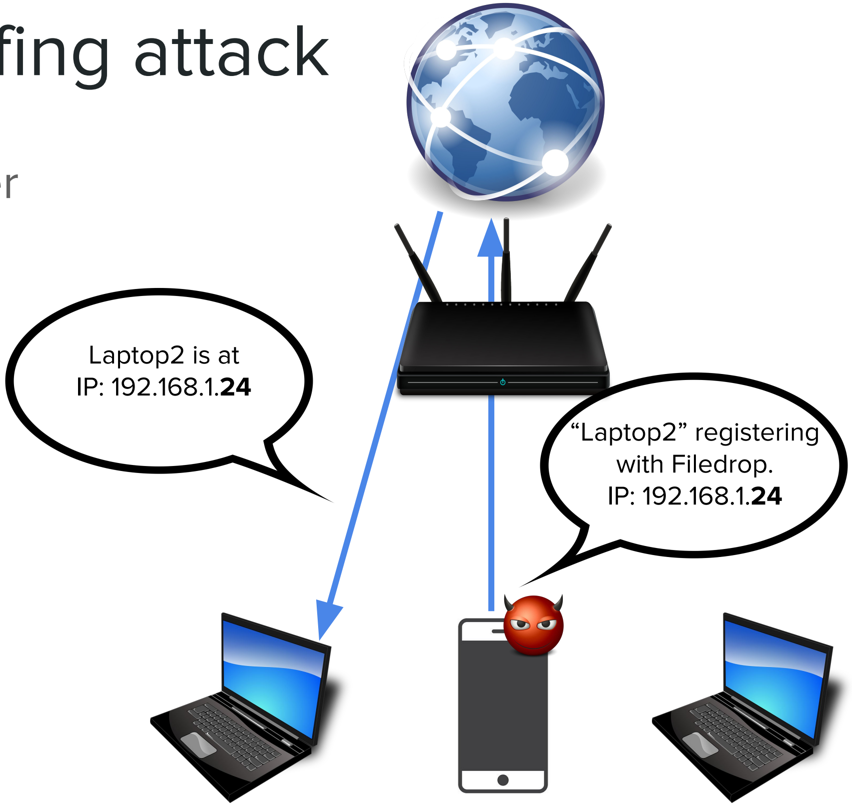


Laptop1 registering with Filedrop. IP: 192.168.1.47
Laptop2 registering with Filedrop. IP: 192.168.1.109

Server registration spoofing attack

Devices use Filedrop server to register for service discovery.

No authentication.



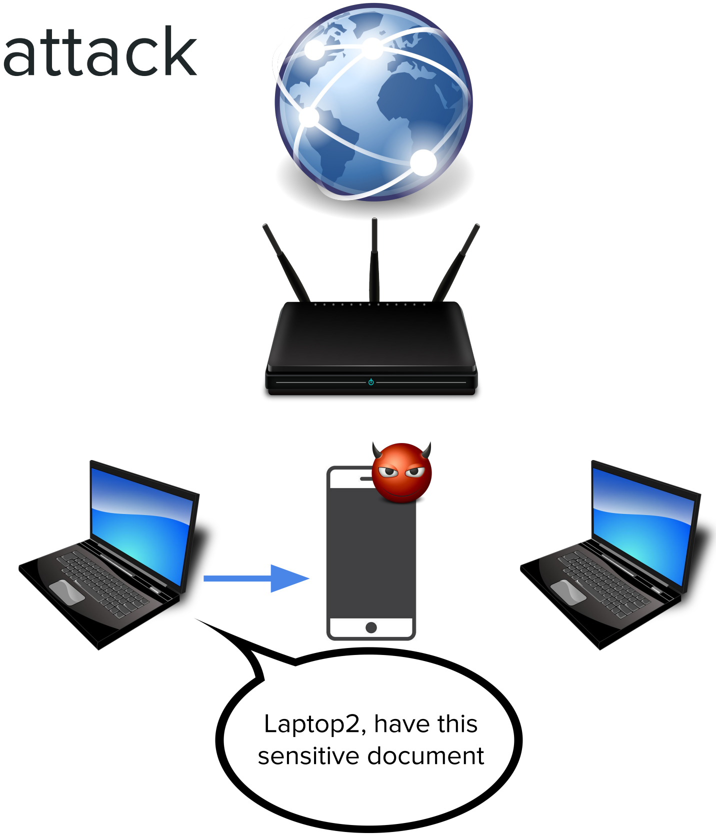
Server registration spoofing attack

Devices use Filedrop server to register for service discovery.

No authentication.

The router cannot introspect on the service discovery process.

Similar to the direct attack scenario, Dreamcatcher is able to block this attack by blocking communication between Laptop1 and the attacker.



Rule categories

1. Direct connection
   - “<Device A> wants to send messages to <Device B>”
   - Unidirectional rules (laptop → printer != printer → laptop)
2. Advertisement
   - “<Device A> wants to advertise itself on your network as <Advertised Name>”
   - User can identify and defend against **Name Poisoning attacks**

Deliberately chose low-granularity rules to maintain usability

Additionally, slight variants of these types, *Broadcast* and *Discovery* rules

Image credits

https://commons.wikimedia.org/wiki/File:%D0%A5%D0%B0%D0%BB%D0%B4%D0%BB%D0%B0%D0%B3%D0%B0.jpg

https://www.techhive.com/article/3004389/connected-home/f-secures-sense-anti-virus-hardware-protects-every-device-in-your-home-from-pcs-to-tvs.html

https://images.techhive.com/images/article/2015/11/fsecuresense-100627152-large.jpg

https://store.nest.com/product/thermostat/T3007ES

https://store.nest.com/assets/images/gallery/thermostat/default/images/1@2x.png?4.22.0.7064-20180523-1

https://commons.wikimedia.org/wiki/File:Openwrt.jpg