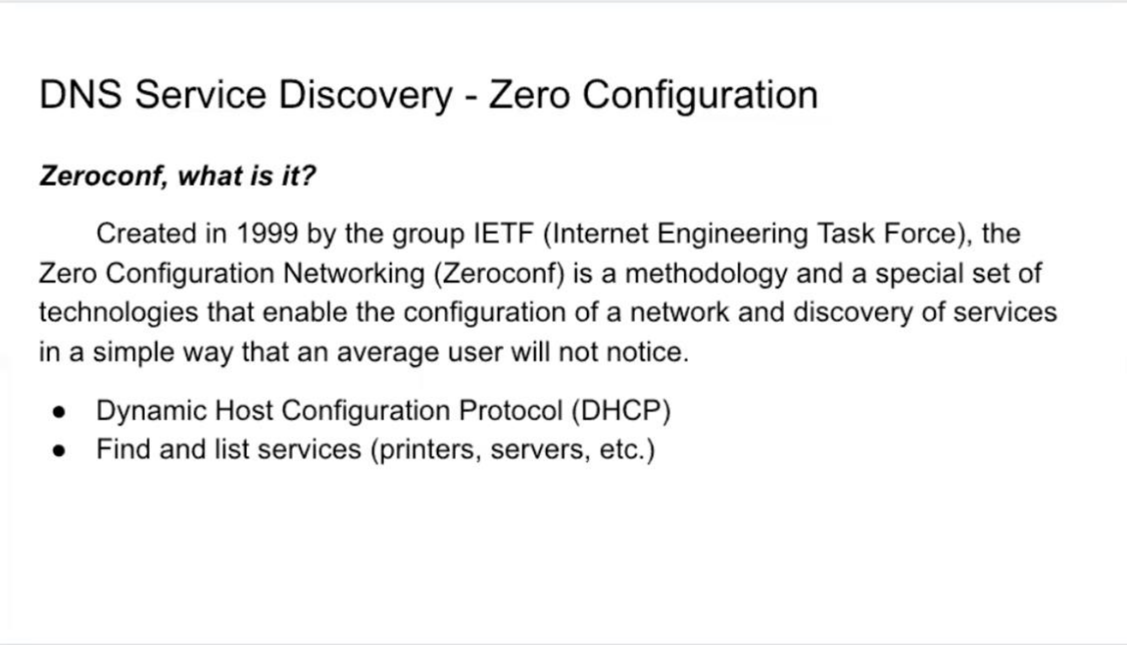
NCAP DISCUSSION GROUP MEETING: 3 MARCH 2021

Name Collision Vulnerabilities

https://docs.google.com/presentation/d/1HNobuSbp2-Er0qefnvsehNmtFbdFvmG9qPOPba-tjzE/edit#slide=id.p

SLIDE 2: DNS Service Discovery - Zero Configuration



DNS Service Discovery - Zero Configuration

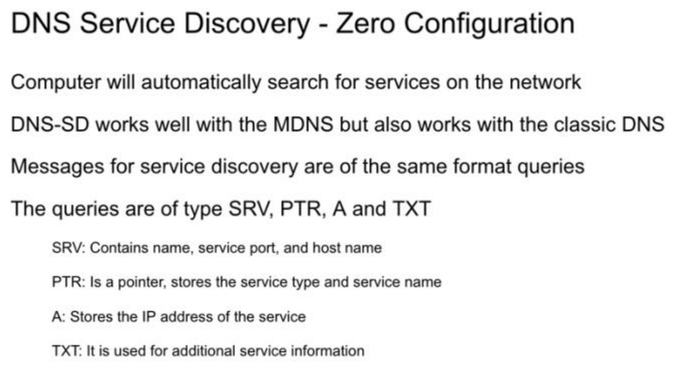
Zeroconf, what is it?

Created in 1999 by the group IETF (Internet Engineering Task Force), the Zero Configuration Networking (Zeroconf) is a methodology and a special set of technologies that enable the configuration of a network and discovery of services in a simple way that an average user will not notice.

- Dynamic Host Configuration Protocol (DHCP)
- Find and list services (printers, servers, etc.)

zero configuration, from the idea back in the late 1990s. when a host joins a network that it can discover the different services or tools available on the network thru DNS service discover, allowing you to identify the printer services there, etc.

Slide 3: cont.



DNS Service Discovery - Zero Configuration

Computer will automatically search for services on the network

DNS-SD works well with the MDNS but also works with the classic DNS

Messages for service discovery are of the same format queries

The queries are of type SRV, PTR, A and TXT

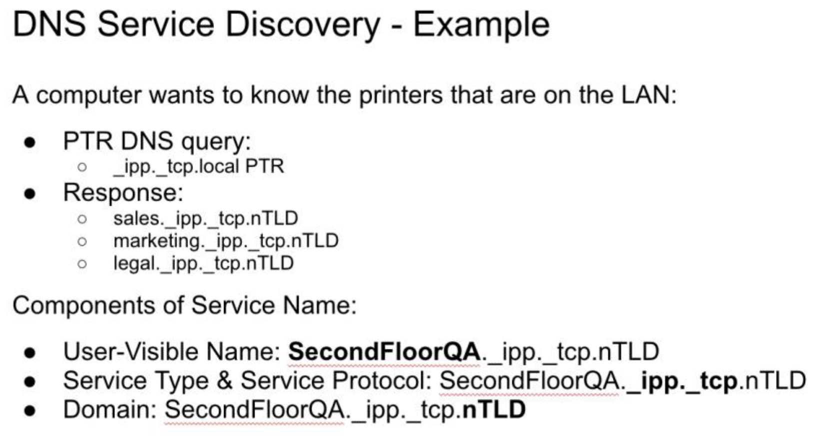
- SRV: Contains name, service port, and host name
- PTR: Is a pointer, stores the service type and service name
- A: Stores the IP address of the service
- TXT: It is used for additional service information

DNS service discovery well it automatically allows you to find those services on the network, but it also works on the traditional DNS as well, and that is the vast majority of these leaks queries

the way that the service discoveries actually work is that they rely on usually four different types of DNS resource record types.

the process usually starts off by the device issue a query, for you know, a US army record saying i'm looking for this phone on this port in the hostname and then it's going to give you back a pointer which will then tell you the actual service type in the service name which then will require an A lookup. To get the IP address port to connect to and then additional queries can be done via txt records to.

Slide 4 &5: DNS Service Discover examples



DNS Service Discovery - Example

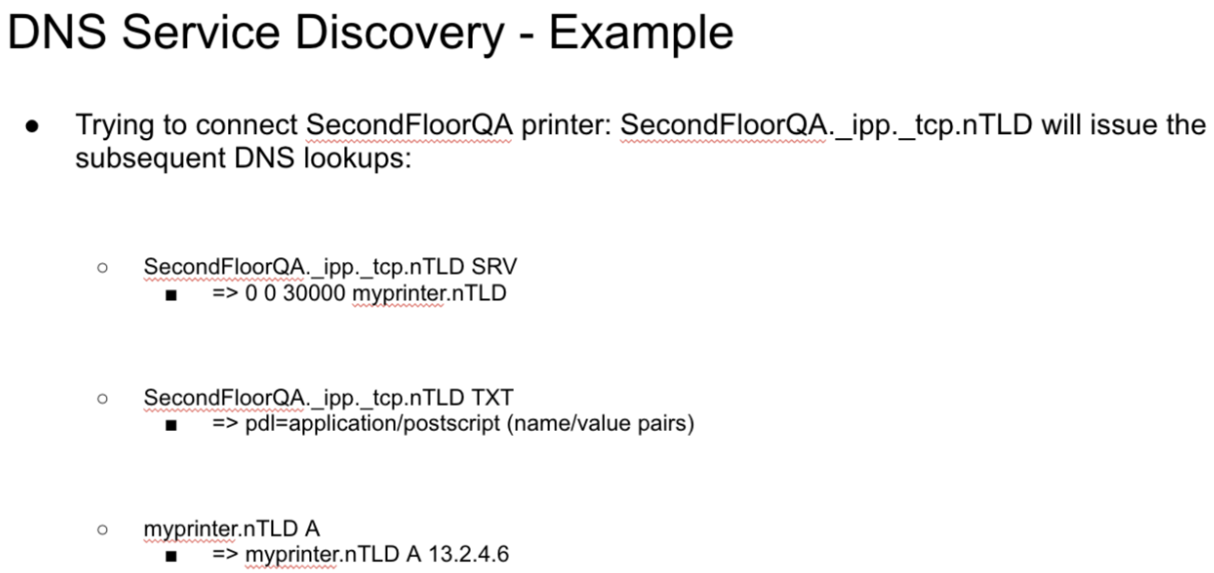
A computer wants to know the printers that are on the LAN:

- PTR DNS query:
  - _ipp._tcp.local PTR
- Response:
  - sales._ipp._tcp.nTLD
  - marketing._ipp._tcp.nTLD
  - legal._ipp._tcp.nTLD

Components of Service Name:

- User-Visible Name: **SecondFloorQA**._ipp._tcp.nTLD
- Service Type & Service Protocol: SecondFloorQA.**_ipp._tcp**.nTLD
- Domain: SecondFloorQA._ipp._tcp.**nTLD**

This is what would happen when a computer wants to know the printers that are on the local area network right the computer is going to send out a DNS service request for underscore IP. in there should be dot underscore tcp underscore your local domain they're here I just chose the word local for the type btr and then you're going to probably expect various responses for the printers that are on the note in this instance,.   you know, maybe there were three different printers you have one for sales, you have one for marketing and one for the legal department. So that first invisible name that's being returned is the actual printers name right so here, you could say like second for QA is the printer me in return for that. And the second portion of that DNS query is the actual type in the protocol so here it's looking for IPP DNS service discovery type and it's going to try and work over the tcp protocol. And then the last component of the service names is the domain right s nd so either this can be obviously hard coded into the configuration of that device or you get it through a suffix search list appendage of for when the domain is going out. And so, that is, you know, probably the string that were relevant in terms of the name collision obviously in terms of looking at calculate risk on a per diem basis. But the rest of it kind of tells us about what kind of services are being used under that particular namespace



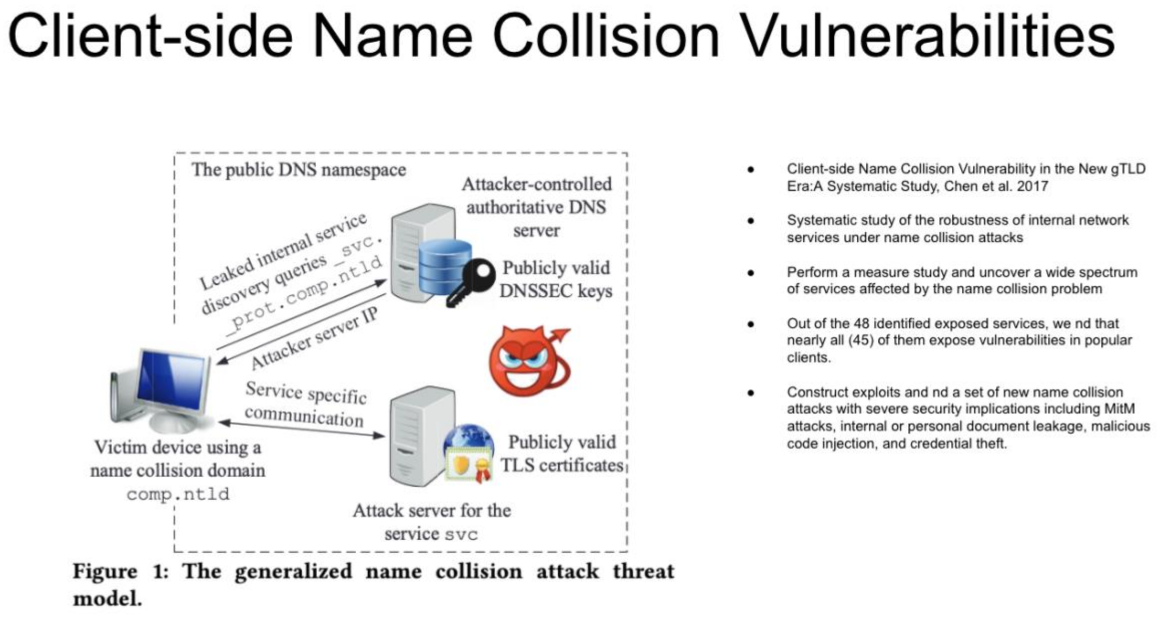
DNS Service Discovery - Example

- Trying to connect SecondFloorQA printer: SecondFloorQA._ipp._tcp.nTLD will issue the subsequent DNS lookups:
  - SecondFloorQA._ipp._tcp.nTLD SRV
    - => 0 0 30000 myprinter.nTLD
  - SecondFloorQA._ipp._tcp.nTLD TXT
    - => pdl=application/postscript (name/value pairs)
  - myprinter.nTLD A
    - => myprinter.nTLD A 13.2.4.6

would see subsequent DNS look ups for the type srp where you're going to get a response for various different properties for that printer including it's another specific name my printer know NTV which then you would also. want to issue another invite issue another txt query where it might pull out name value pairs that are stored inside the txt record for that, and then, finally, it will do the traditional a or upon a look up to get the IP address to actually connect on that.And so, this is where you know, obviously in an inclusion scenario where these names were supposed to reside in a confined network, but now are leaking out into the global DNS.

If that domain becomes available, it is possible that someone could answer those questions and have these zero configuration requests return to malicious servers that user wasn't aware of, or intended them to go to.

Slide 6: Client-side Name Collision Vulnerabilities


The public DNS namespace
Attacker-controlled authoritative DNS server
Leaked internal service discovery queries _svc._prot.comp.ntld
Attacker server IP
Publicly valid DNSSEC keys
Service specific communication
Victim device using a name collision domain comp.ntld
Publicly valid TLS certificates
Attack server for the service svc

Client-side Name Collision Vulnerabilities

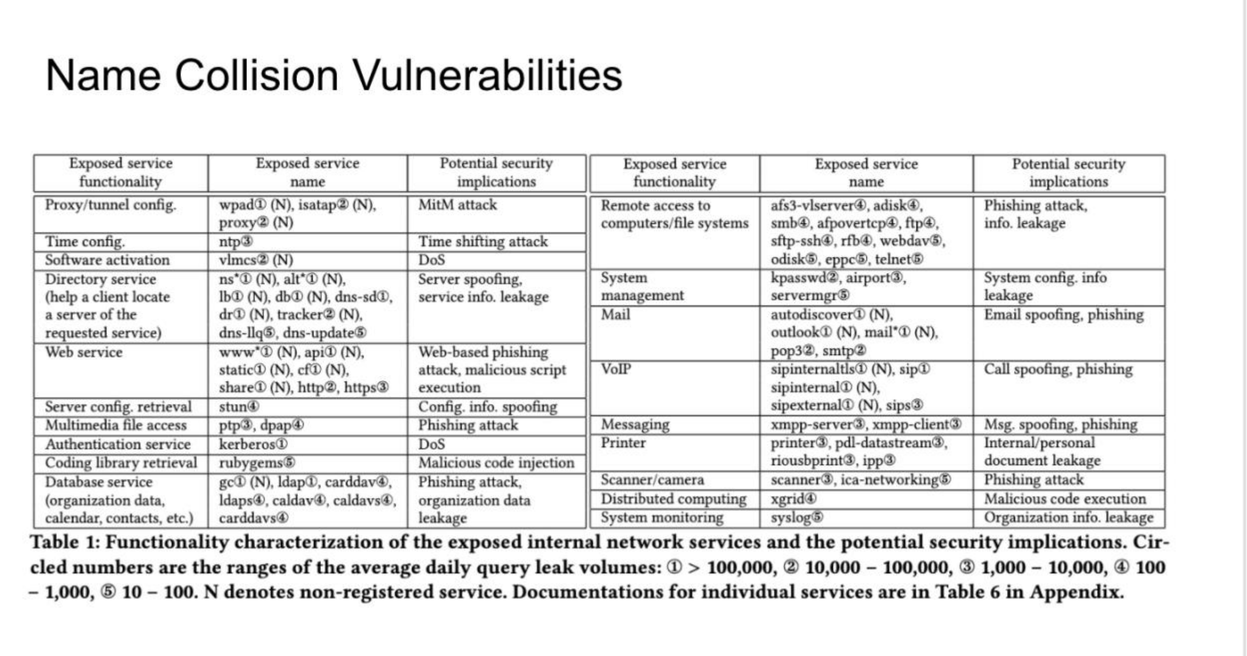
Figure 1: The generalized name collision attack threat model.

- Client-side Name Collision Vulnerability in the New gTLD Era:A Systematic Study, Chen et al. 2017
- Systematic study of the robustness of internal network services under name collision attacks
- Perform a measure study and uncover a wide spectrum of services affected by the name collision problem
- Out of the 48 identified exposed services, we find that nearly all (45) of them expose vulnerabilities in popular clients.
- Construct exploits and find a set of new name collision attacks with severe security implications including MitM attacks, internal or personal document leakage, malicious code injection, and credential theft.

hese are more for actual configuration as services, the chromium one is actually a test on like connectivity or or Internet redirection right that the DNS resolve or isn't really honoring non existent domains, and it was trying to detect we auto redirection to something else right.

when we're saying the vulnerability, the vulnerability only happens if the tlc is delegated and someone registers, the exact second level name that.would be that would actually be a collision right so it's it's not by delegation to the to the necessary necessarily alone it's by delegation of the to the plus someone registering the malicious person registering a domain.

Slide 7 & 8 : Name Collision Vulnerabilities

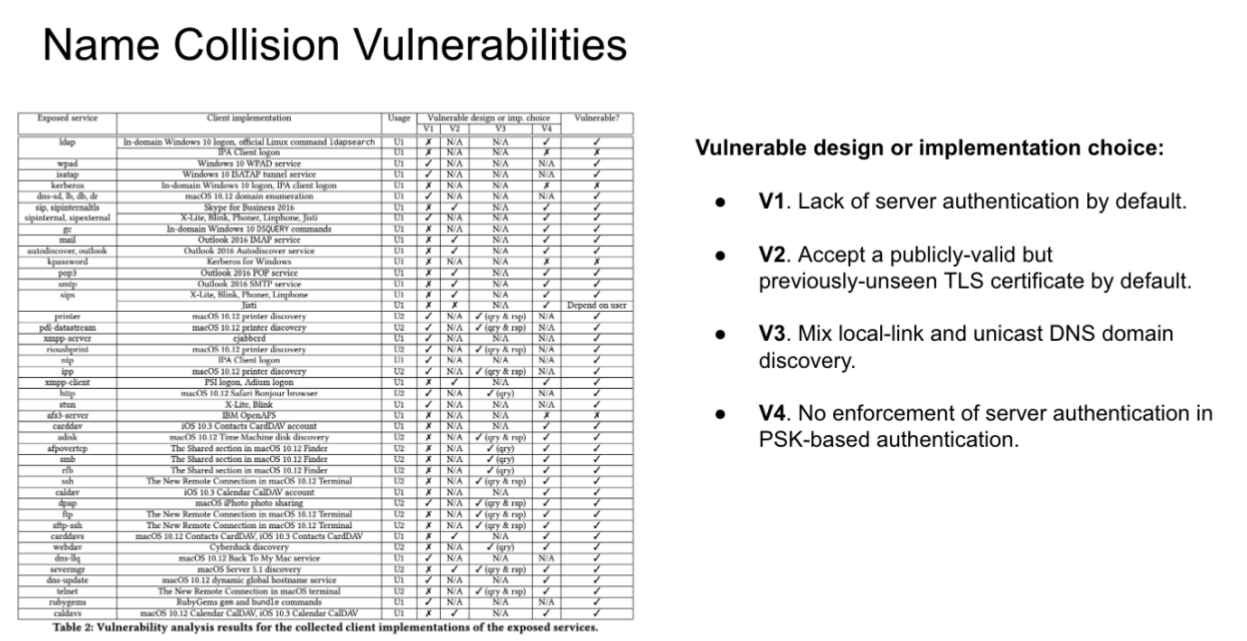


Name Collision Vulnerabilities

Exposed service functionality	Exposed service name	Potential security implications
Proxy/tunnel config.	wpad① (N), isatap② (N), proxy② (N)	MitM attack
Time config.	ntp③	Time shifting attack
Software activation	vlmcs② (N)	DoS
Directory service (help a client locate a server of the requested service)	ns*① (N), alt*① (N), lb① (N), db① (N), dns-sd①, dr① (N), tracker② (N), dns-llq⑤, dns-update⑤	Server spoofing, service info. leakage
Web service	www*① (N), api① (N), static① (N), cf① (N), share① (N), http②, https③	Web-based phishing attack, malicious script execution
Server config. retrieval	stun④	Config. info. spoofing
Multimedia file access	ptp③, dpap④	Phishing attack
Authentication service	kerberos①	DoS
Coding library retrieval	rubygems⑤	Malicious code injection
Database service (organization data, calendar, contacts, etc.)	gc① (N), ldap①, carddav④, ldaps④, caldav④, caldavs④, carddavs④	Phishing attack, organization data leakage
Remote access to computers/file systems	afs3-vlserver④, adisk④, smb④, afpovertcp④, ftp④, sftp-ssh④, rfb④, webdav⑤, odisk⑤, eppc⑤, telnet⑤	Phishing attack, info. leakage
System management	kpasswd②, airport③, servermgr⑤	System config. info leakage
Mail	autodiscover① (N), outlook① (N), mail*① (N), pop3②, smtp②	Email spoofing, phishing
VoIP	sipinternaltls① (N), sip① sipinternal① (N), sipexternal① (N), sips③	Call spoofing, phishing
Messaging	xmpp-server③, xmpp-client③	Msg. spoofing, phishing
Printer	printer③, pdl-datastream③, riousbprint③, ipp③	Internal/personal document leakage
Scanner/camera	scanner③, ica-networking⑤	Phishing attack
Distributed computing	xgrid④	Malicious code execution
System monitoring	syslog⑤	Organization info. leakage

Table 1: Functionality characterization of the exposed internal network services and the potential security implications. Circled numbers are the ranges of the average daily query leak volumes: ① > 100,000, ② 10,000 – 100,000, ③ 1,000 – 10,000, ④ 100 – 1,000, ⑤ 10 – 100. N denotes non-registered service. Documentations for individual services are in Table 6 in Appendix.

We can get a sense of what some of these vulnerabilities are as well as the specific service name that was used inside the queue name. For that specific DNS service discovery so some of the things like probably the most stone, I would say, dangerous would be the man in the middle attacks, where you have things like the w pad ice attack.



Name Collision Vulnerabilities

Exposed service	Client implementation	Usage	Vulnerable design or imp. choice				Vulnerable?
			V1	V2	V3	V4	
ldap	In-domain Windows 10 logon, official Linux command ldapsearch	U1	✗	N/A	N/A	✓	✓
	IPA Client logon	U1	✗	N/A	N/A	✗	✗
wpad	Windows 10 WPAD service	U1	✓	N/A	N/A	N/A	✓
isatap	Windows 10 ISATAP tunnel service	U1	✓	N/A	N/A	N/A	✓
kerberos	In-domain Windows 10 logon, IPA client logon	U1	✗	N/A	N/A	✗	✗
dns-sd, lb, db, dr	macOS 10.12 domain enumeration	U1	✓	N/A	N/A	N/A	✓
sip, sipinternaltls sipinternal, sipexternal	Skype for Business 2016	U1	✗	✓	N/A	✓	✓
	X-Lite, Blink, Phoner, Linphone, Jisti	U1	✓	N/A	N/A	✓	✓
gc	In-domain Windows 10 DSQUERY commands	U1	✗	N/A	N/A	✓	✓
mail	Outlook 2016 IMAP service	U1	✗	✓	N/A	✓	✓
autodiscover, outlook	Outlook 2016 Autodiscover service	U1	✗	✓	N/A	✓	✓
kpassword	Kerberos for Windows	U1	✗	N/A	N/A	✗	✗
pop3	Outlook 2016 POP service	U1	✗	✓	N/A	✓	✓
smtp	Outlook 2016 SMTP service	U1	✗	✓	N/A	✓	✓
sips	X-Lite, Blink, Phoner, Linphone	U1	✗	✓	N/A	✓	✓
	Jisti	U1	✗	✗	N/A	✓	Depend on user
printer	macOS 10.12 printer discovery	U2	✓	N/A	✓ (qry & rsp)	N/A	✓
pdl-datastream	macOS 10.12 printer discovery	U2	✓	N/A	✓ (qry & rsp)	N/A	✓
xmpp-server	cjabberd	U1	✓	N/A	N/A	N/A	✓
riousbprint	macOS 10.12 printer discovery	U2	✓	N/A	✓ (qry & rsp)	N/A	✓
ntp	IPA Client logon	U1	✓	N/A	N/A	N/A	✓
ipp	macOS 10.12 printer discovery	U2	✓	N/A	✓ (qry & rsp)	N/A	✓
xmpp-client	PSI logon, Adium logon	U1	✗	✓	N/A	✓	✓
http	macOS 10.12 Safari Bonjour browser	U2	✓	N/A	✓ (qry)	N/A	✓
stun	X-Lite, Blink	U1	✓	N/A	N/A	N/A	✓
afs3-server	IBM OpenAFS	U1	✗	N/A	N/A	✗	✗
carddav	iOS 10.3 Contacts CardDAV account	U1	✗	N/A	N/A	✓	✓
adisk	macOS 10.12 Time Machine disk discovery	U2	✗	N/A	✓ (qry & rsp)	✓	✓
afpovertcp	The Shared section in macOS 10.12 Finder	U2	✗	N/A	✓ (qry)	✓	✓
smb	The Shared section in macOS 10.12 Finder	U2	✗	N/A	✓ (qry)	✓	✓
rfb	The Shared section in macOS 10.12 Finder	U2	✗	N/A	✓ (qry)	✓	✓
ssh	The New Remote Connection in macOS 10.12 Terminal	U2	✗	N/A	✓ (qry & rsp)	✓	✓
caldav	iOS 10.3 Calendar CalDAV account	U1	✗	N/A	N/A	✓	✓
dpap	macOS iPhoto photo sharing	U2	✓	N/A	✓ (qry & rsp)	✓	✓
ftp	The New Remote Connection in macOS 10.12 Terminal	U2	✗	N/A	✓ (qry & rsp)	✓	✓
sftp-ssh	The New Remote Connection in macOS 10.12 Terminal	U2	✗	N/A	✓ (qry & rsp)	✓	✓
carddavs	macOS 10.12 Contacts CardDAV, iOS 10.3 Contacts CardDAV	U1	✗	✓	N/A	✓	✓
webdav	Cyberduck discovery	U2	✗	N/A	✓ (qry)	✓	✓
dns-llq	macOS 10.12 Back To My Mac service	U1	✓	N/A	N/A	N/A	✓
severmgr	macOS Server 5.1 discovery	U2	✗	✓	✓ (qry & rsp)	✓	✓
dns-update	macOS 10.12 dynamic global hostname service	U1	✓	N/A	N/A	✓	✓
telnet	The New Remote Connection in macOS terminal	U2	✗	N/A	✓ (qry & rsp)	✓	✓
rubygems	RubyGems gem and bundle commands	U1	✓	N/A	N/A	N/A	✓
caldavs	macOS 10.12 Calendar CalDAV, iOS 10.3 Calendar CalDAV	U1	✗	✓	N/A	✓	✓

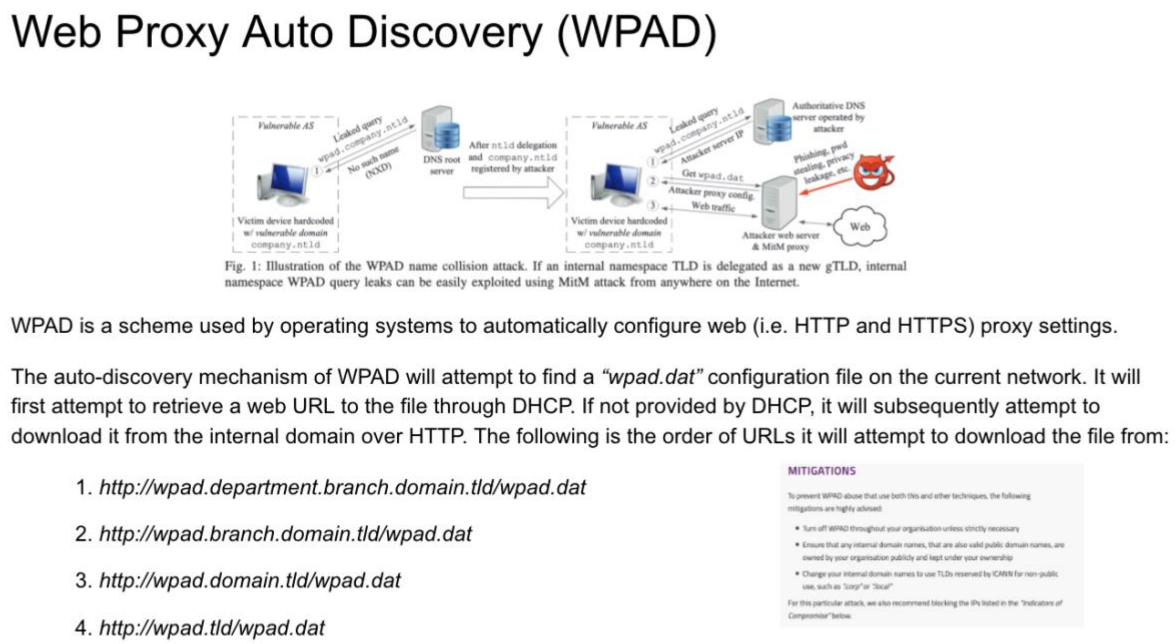
Table 2: Vulnerability analysis results for the collected client implementations of the exposed services.

Vulnerable design or implementation choice:

- **V1**. Lack of server authentication by default.
- **V2**. Accept a publicly-valid but previously-unseen TLS certificate by default.
- **V3**. Mix local-link and unicast DNS domain discovery.
- **V4**. No enforcement of server authentication in PSK-based authentication.

the vast majority of these 45 different service service discovery protocols all were due to poor authentication and

Slide 9: Web Proxy Auto Discovery (WPAD)



Web Proxy Auto Discovery (WPAD)

Vulnerable AS
Leaked query
wpad.company.ntld
No such name (NXD)
1
DNS root server
Victim device hardcoded w/ vulnerable domain company.ntld
After ntld delegation and company.ntld registered by attacker
Vulnerable AS
Leaked query
wpad.company.ntld
Attacker server IP
Authoritative DNS server operated by attacker
1
2
3
Get wpad.dat
Attacker proxy config.
Web traffic
Phishing, pwd stealing, privacy leakage, etc.
Victim device hardcoded w/ vulnerable domain company.ntld
Attacker web server & MitM proxy
Web

Fig. 1: Illustration of the WPAD name collision attack. If an internal namespace TLD is delegated as a new gTLD, internal namespace WPAD query leaks can be easily exploited using MitM attack from anywhere on the Internet.

WPAD is a scheme used by operating systems to automatically configure web (i.e. HTTP and HTTPS) proxy settings.

The auto-discovery mechanism of WPAD will attempt to find a *"wpad.dat"* configuration file on the current network. It will first attempt to retrieve a web URL to the file through DHCP. If not provided by DHCP, it will subsequently attempt to download it from the internal domain over HTTP. The following is the order of URLs it will attempt to download the file from:

1. *http://wpad.department.branch.domain.tld/wpad.dat*
2. *http://wpad.branch.domain.tld/wpad.dat*
3. *http://wpad.domain.tld/wpad.dat*
4. *http://wpad.tld/wpad.dat*

MITIGATIONS

To prevent WPAD abuse that use both this and other techniques, the following mitigations are highly advised:

- Turn off WPAD throughout your organisation unless strictly necessary
- Ensure that any internal domain names, that are also valid public domain names, are owned by your organisation publicly and kept under your ownership
- Change your internal domain names to use TLDs reserved by ICANN for non-public use, such as "corp" or "local"

For this particular attack, we also recommend blocking the IPs listed in the "Indicators of Compromise" below.