Zero Trust & The Flaming Sword of Justice

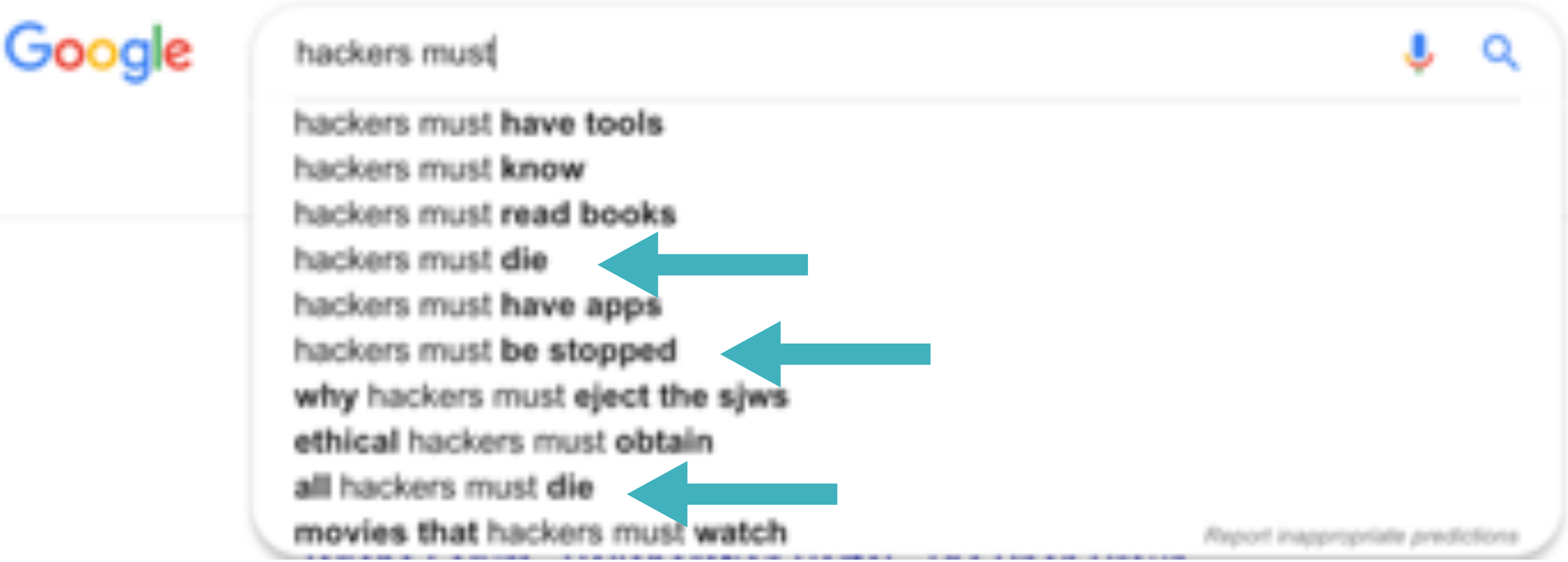
Dave Lewis, Global Advisory CISO

Duo Security, Cisco

@gattaca

#WHOAMI

Dave Lewis, Global Advisory CISO


Google
hackers must
hackers must have tools
hackers must know
hackers must read books
hackers must die
hackers must have apps
hackers must be stopped
why hackers must eject the sjws
ethical hackers must obtain
all hackers must die
movies that hackers must watch
Report inappropriate predictions


CISCO

Adaptive Multi-Factor Authentication (MFA)

User and device trust for every application


Our mission
is to protect
your mission.

What is Zero Trust?

- Where/how/when trust is decided has changed
- Must continuously verify
- Assume all networks are hostile
- This is not a “rip & replace” conversation

ZTN

DUO


It’s On Fire Yo!


WORKED FINE IN DEV
OPS PROBLEM NOW

What is Zero Trust?

- 2004(ish) - Jericho Forum
- 2010 - John Kindervag, coined term ‘Zero Trust’
- 2014 - Google BeyondCorp
- 2017 - O’Reilly Zero Trust Networks

Change in IT Landscape

With a shift to cloud and mobile-first world, a new approach to security is needed.

Cloud **Mobile** **Anywhere**

Shift to Zero Trust Trends

Security/IT teams need to enable user access, from any device and anywhere, while preventing breaches.

Enable Multi- Cloud Access

- Cloud infrastructure
- SaaS apps

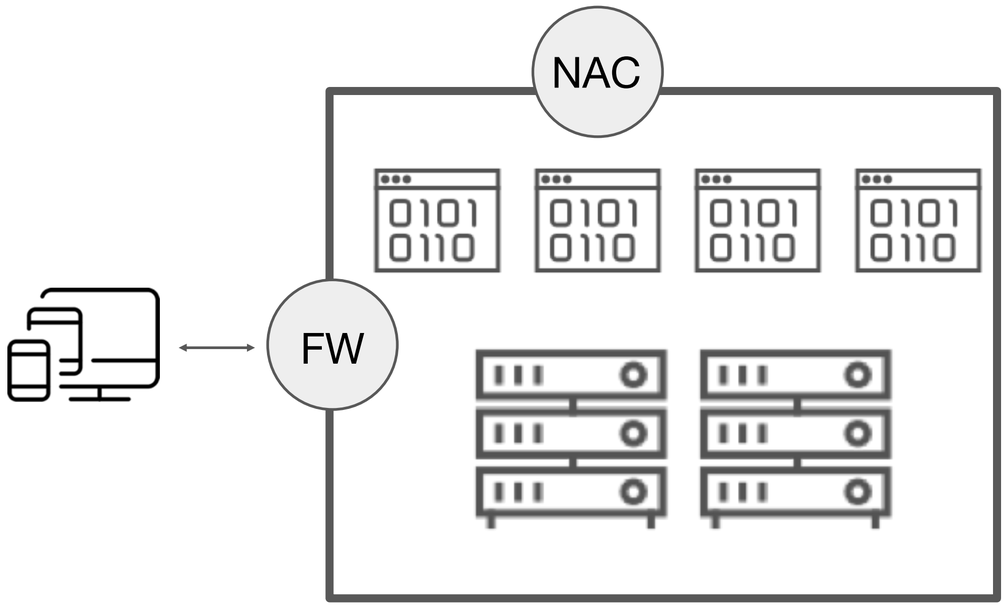
Enable BYOD

- Work from anywhere using personal devices
- Third-parties, contractors, etc.

Breach Prevention

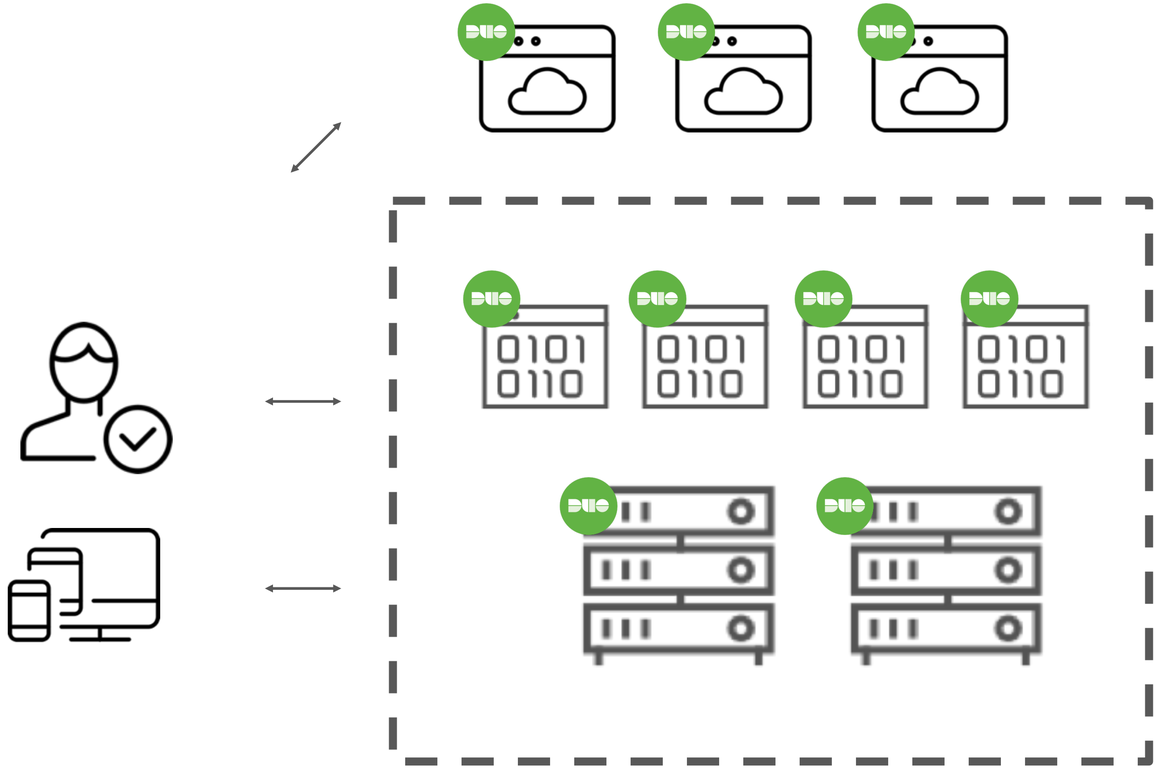
- Compromised credentials
- Phishing

Traditional perimeter approach



Focus on securing **access to the network** by inspecting the device.

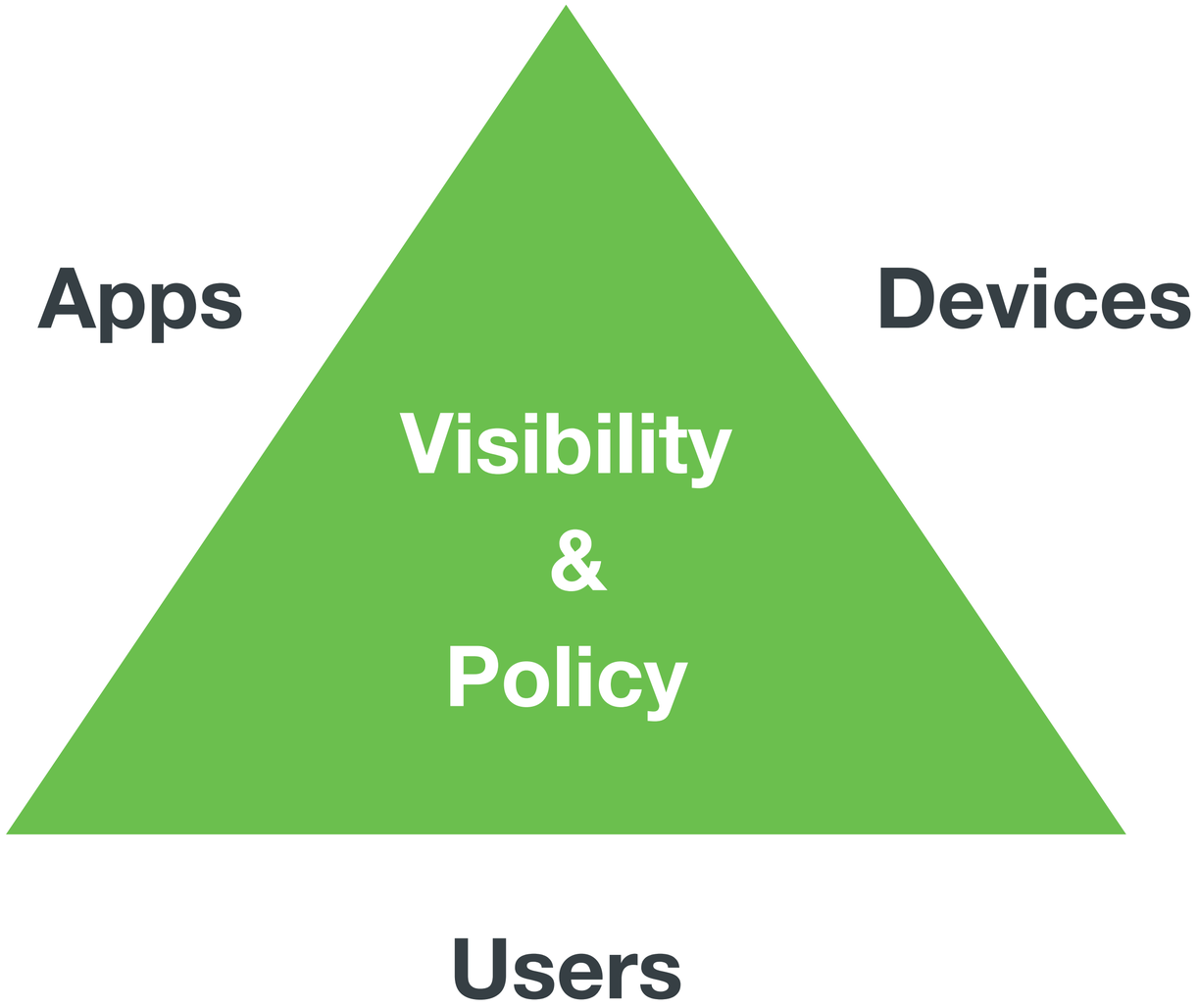
“Zero Trust” approach



Focus on securing **access to the application** by **verifying the user**, **inspecting the device** and **context**

Coverage for all apps including SaaS apps outside the corporate network.

The Magic Triad



How Do You Establish Trust Easily and Effectively?

Castles Don’t Scale

Don’t trust something just because it’s on the “inside” of your firewall

Is the password…password?

No!! Now go away, or I shall taunt you a second time!

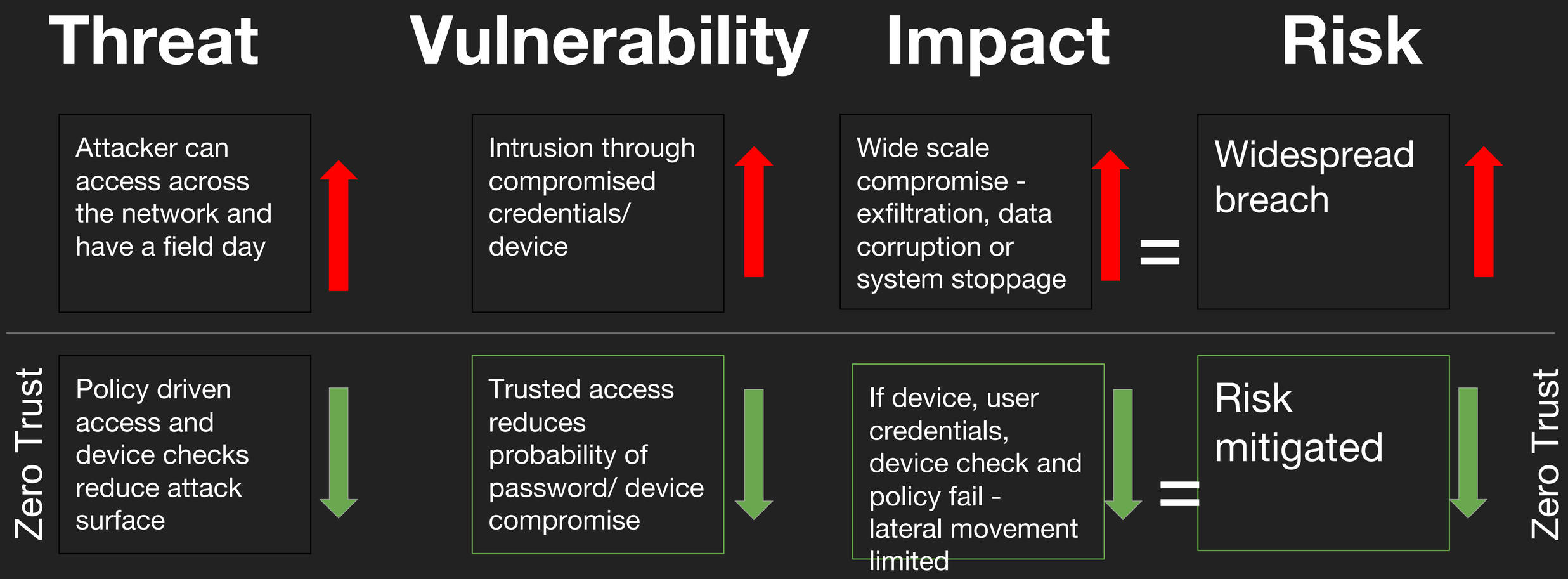
Lessons From History

The sack of Rome in 410 AD


Remember when you only had to outrun the other hiker?

Now there’s more than enough bear to go around

Reducing the Risk



The Flaming Sword of Justice

Data Breaches

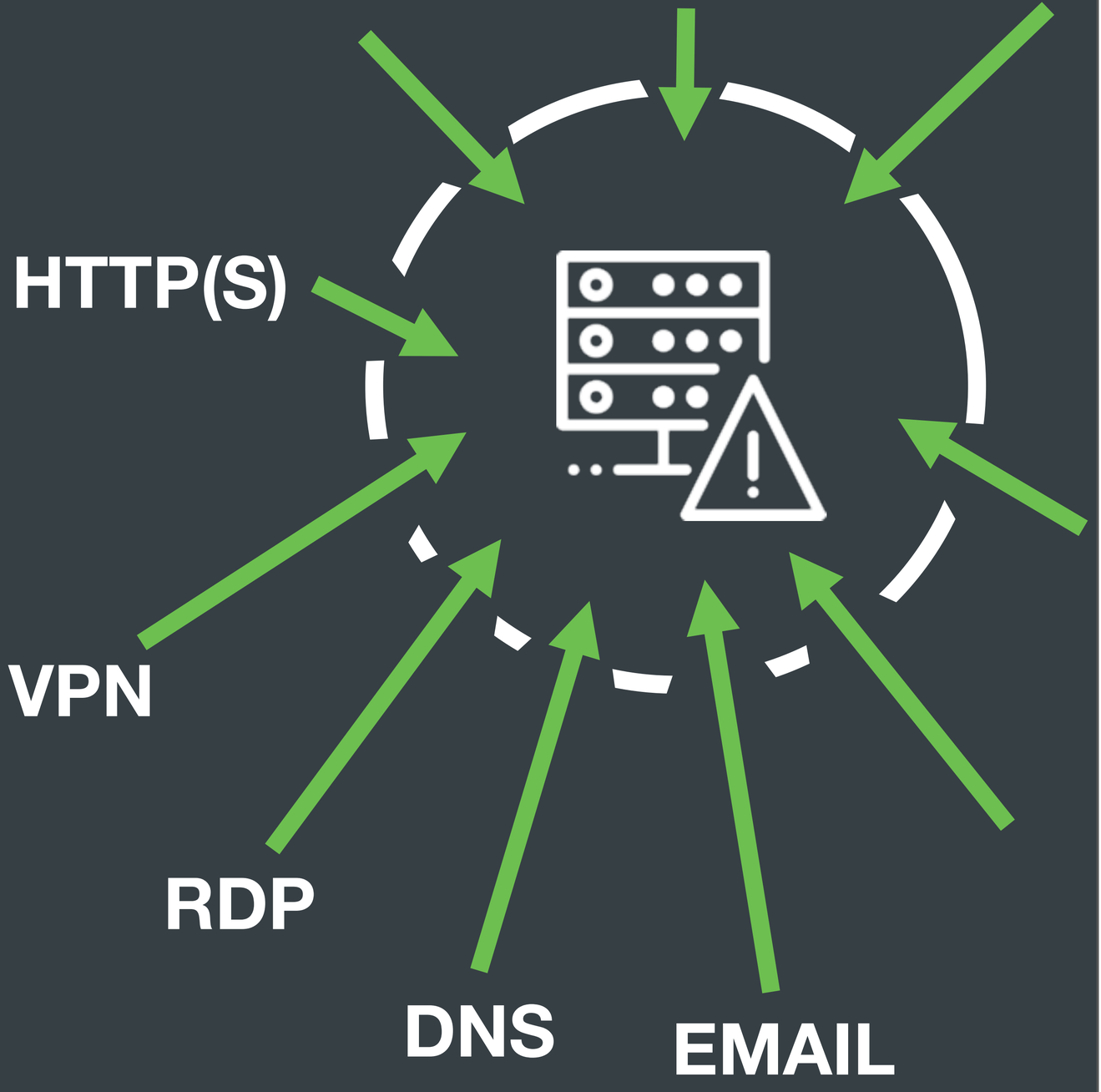
81%

Of breaches involve stolen or weak **credentials**

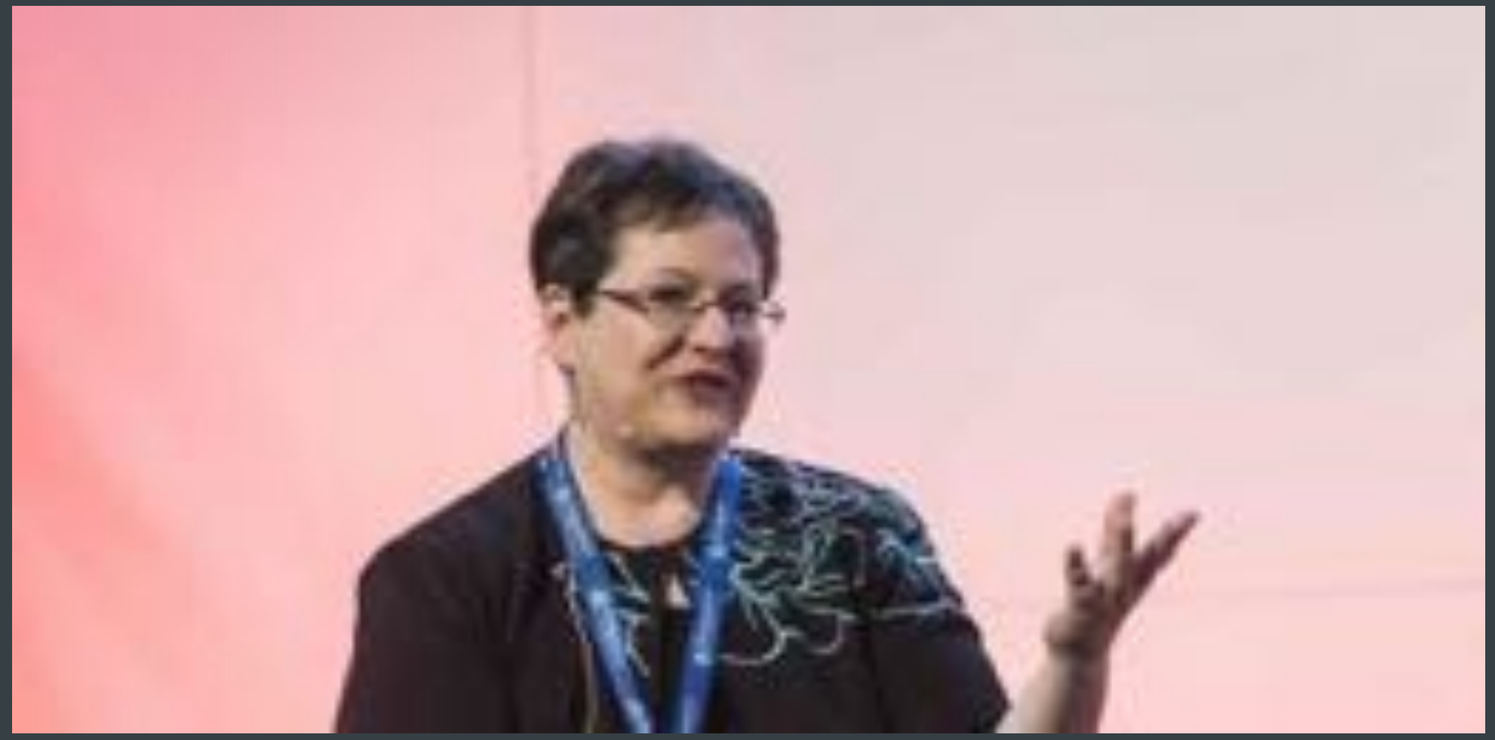
70%

Of breaches involve compromised **devices**

Porous Perimeter




Are You
My
Perimeter?



“The perimeter is anywhere an access decision is being made.”

New Perimeter

Remote Employees

Hybrid Cloud

Old Perimeter

Traditional Network:
Endpoints, On-site Users, Servers, Apps

Cloud Applications

Personal Devices

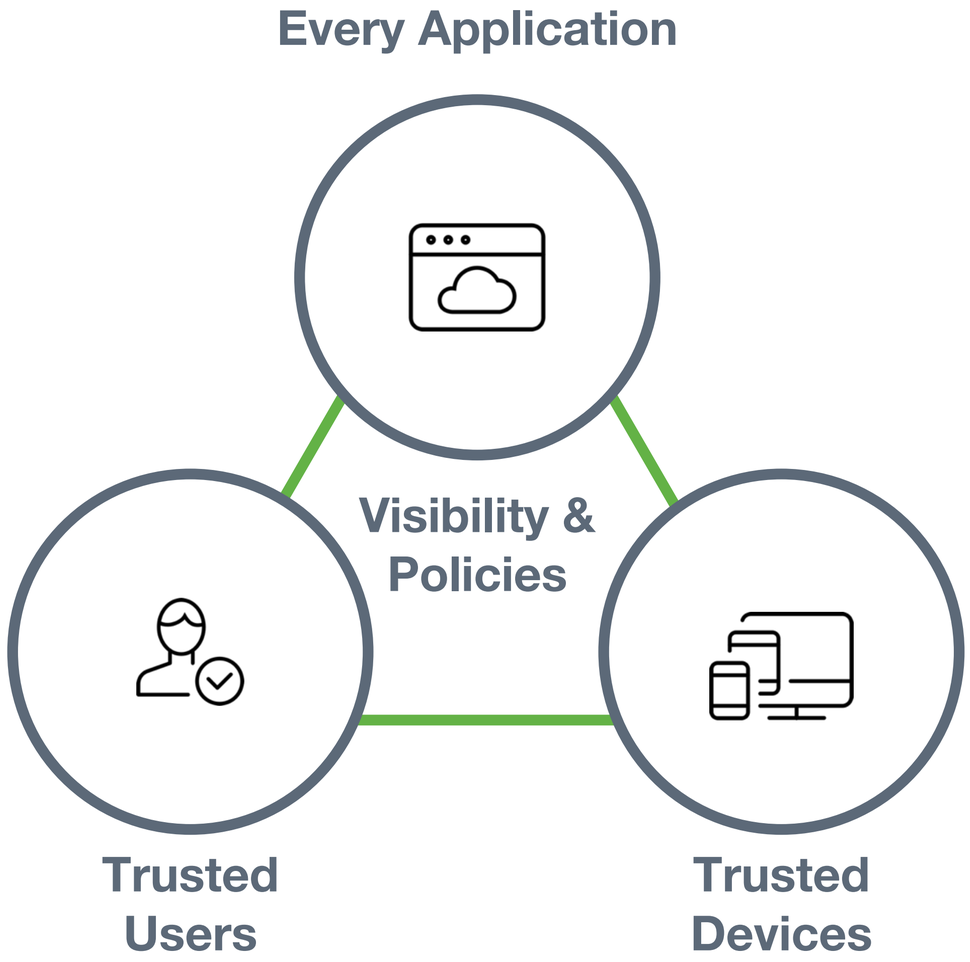
Mobile Devices

Vendors & Contractors

DUO

Zero-Trust Model: Focus on Access

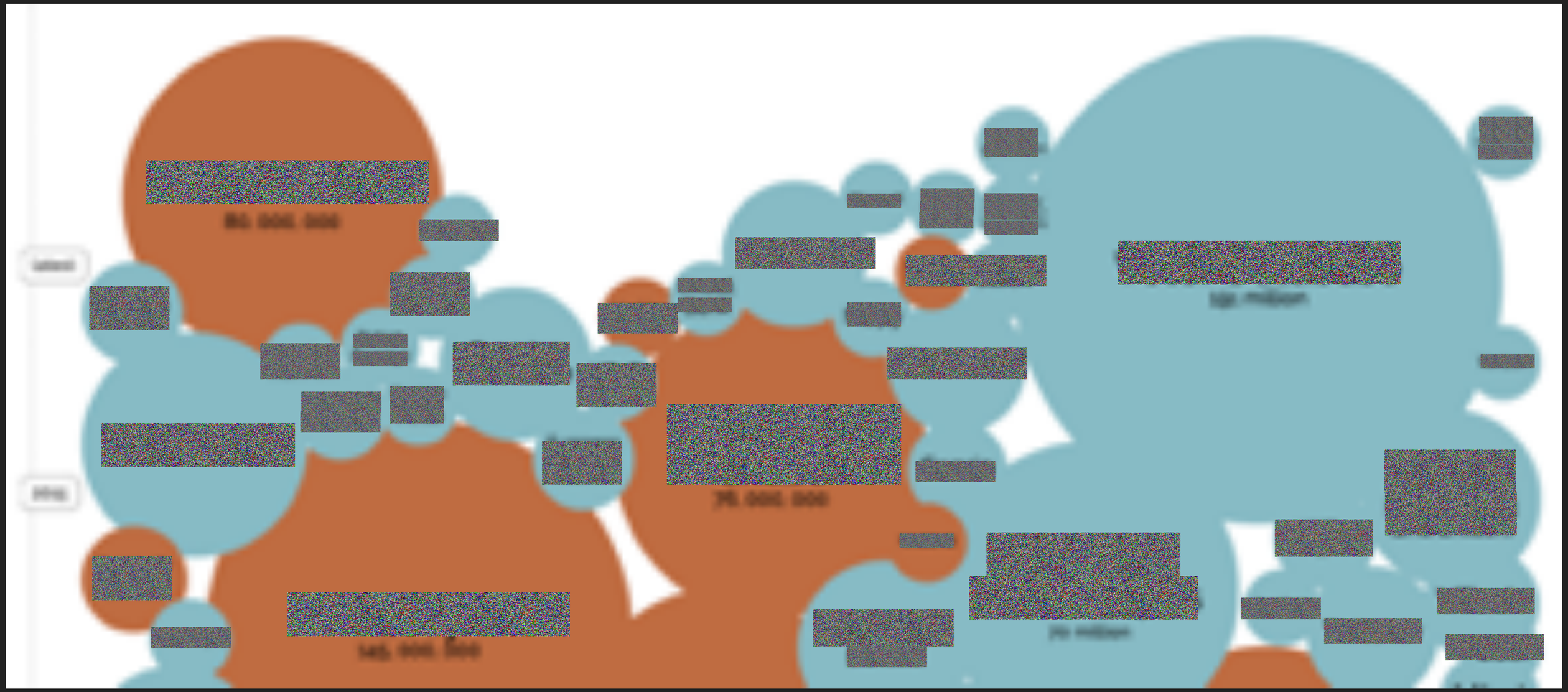
Protect organizations by verifying the identity of **users** and the health of their **devices** before connecting to the **applications** they need.



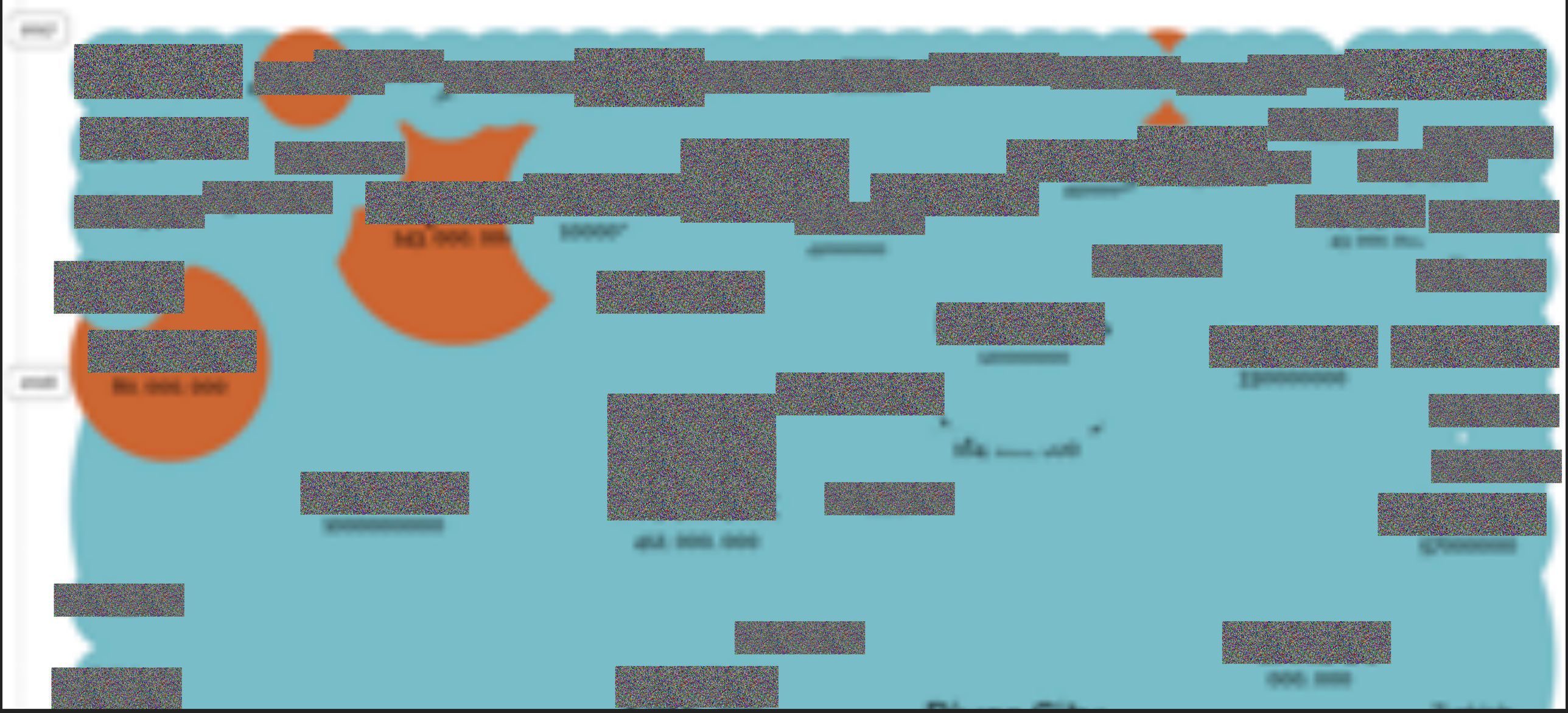
The Summer of Breach 2012

Site Breached	Users Affected	Link	Confirmed
Yahoo	453,000	CNN	Yes
Formspring	420,000	Securityweek	Yes
Phandroid	1,000,000	Securityweek	Yes
Billabong	21,485	IT News AU	Yes
Nvidia	800	PCWorld	Yes
LinkedIn	6,460,000	Globe and Mail	Yes
eHarmony	1,500,000	ZDNet	Yes
Consumerist	TBD	Consumerist	Yes/TBD

Been There…



FOUR YEARS AGO . . .



THREE YEARS AGO . . . AND NOW

What’s Open In USA?

182,698,085

Systems Vulnerable to Eternal Blue

1,562

Systems Vulnerable To Heartbleed

18,205

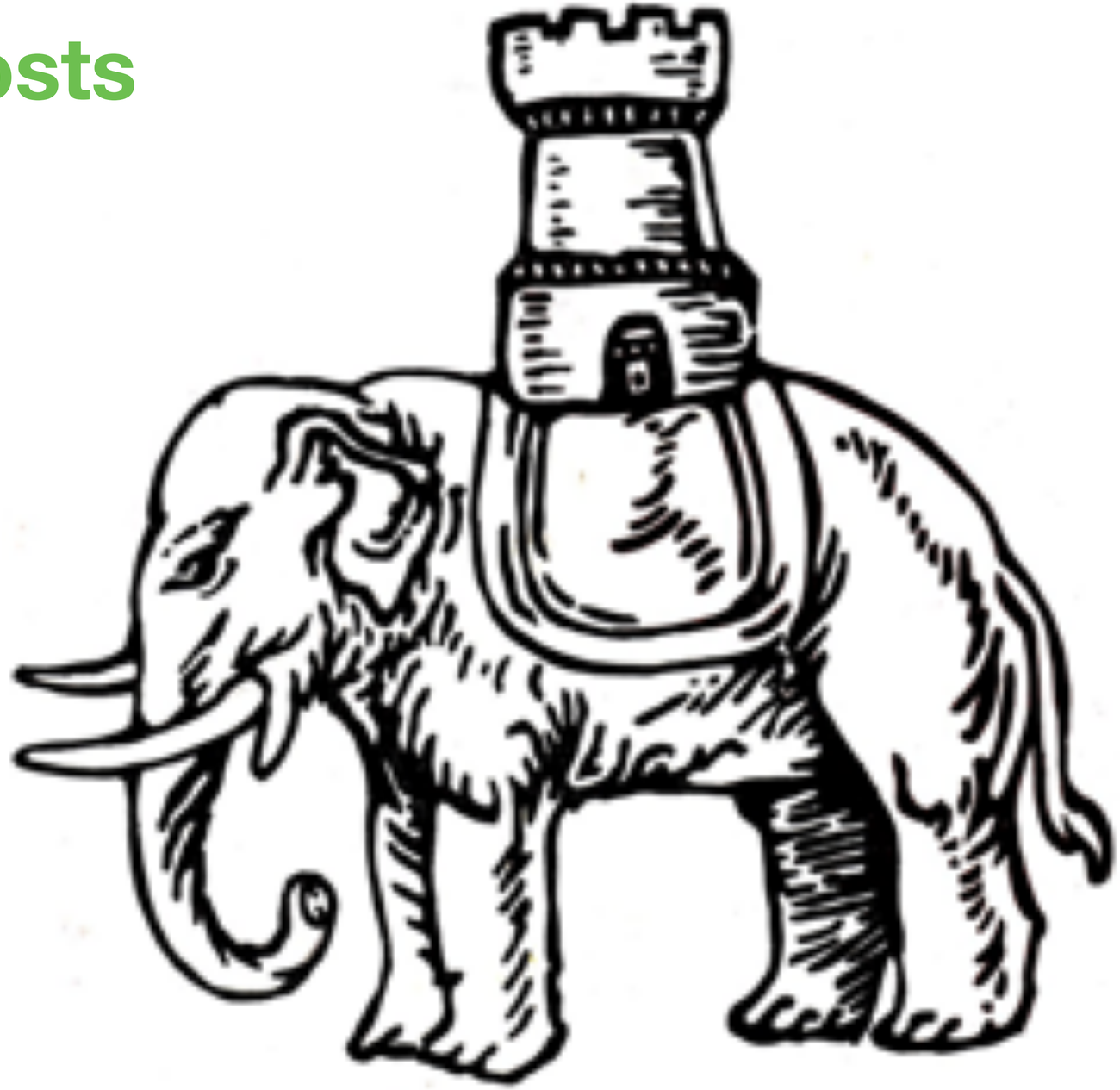
Systems Vulnerable to BlueKeep

38,999

Trusted Access Security, Value Proposition

- Devaluation of stolen credentials
- Low hanging fruit sours.
- Complicates lateral movement through uniform security policy.
- Attackers have to work that much harder.

Bastion Hosts



From DMZ To The Soft Chewy Centre

Setting Expectations

A Game of Increments

Determining Priorities

How do you stop attacks that use stolen (yet legitimate) credentials?

How do you prevent devices with poor security hygiene from accessing critical apps?

Security Best Practices

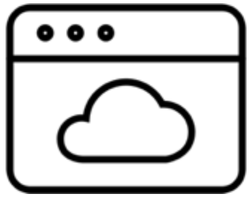
Policies Are Unique to Each User and Device

Verify Your Users

- Strong Authentication
- Intuitive Authentication
- User Risk Assessment

Verify Their Devices

- Up-to-date Devices
- Well-configured Devices
- Managed Devices
- Device Authentication



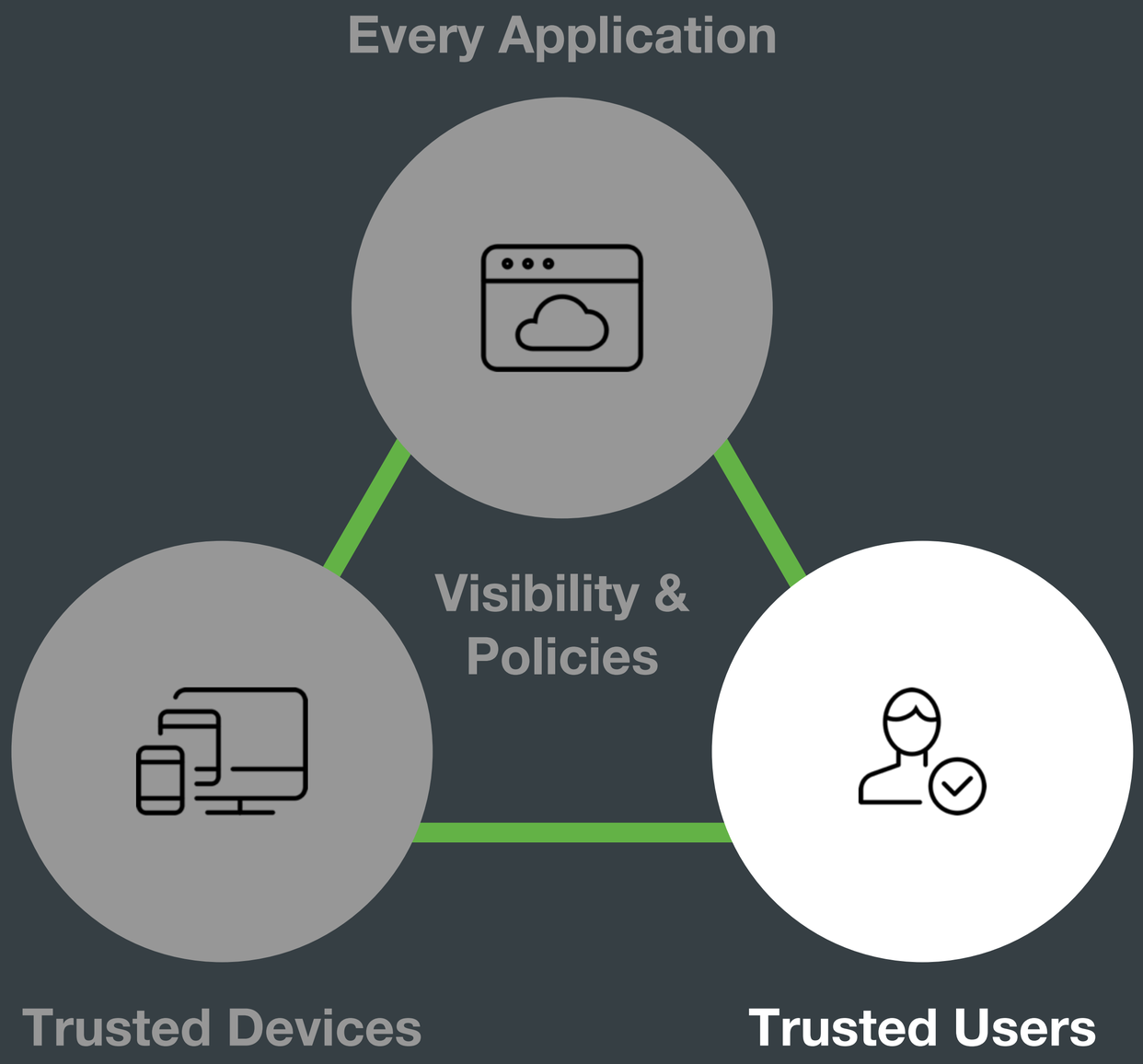
Protect Every Application

- All Cloud Apps
- All On-Prem Apps
- Consistent End User Experience & Security

Security Beyond the Perimeter

Trusted Users:

- ➔Allow only known users
- ➔Verify identity with strong authentication
- ➔Regardless of location
- ➔On every access
- ➔For every application



Example: Stolen Credentials

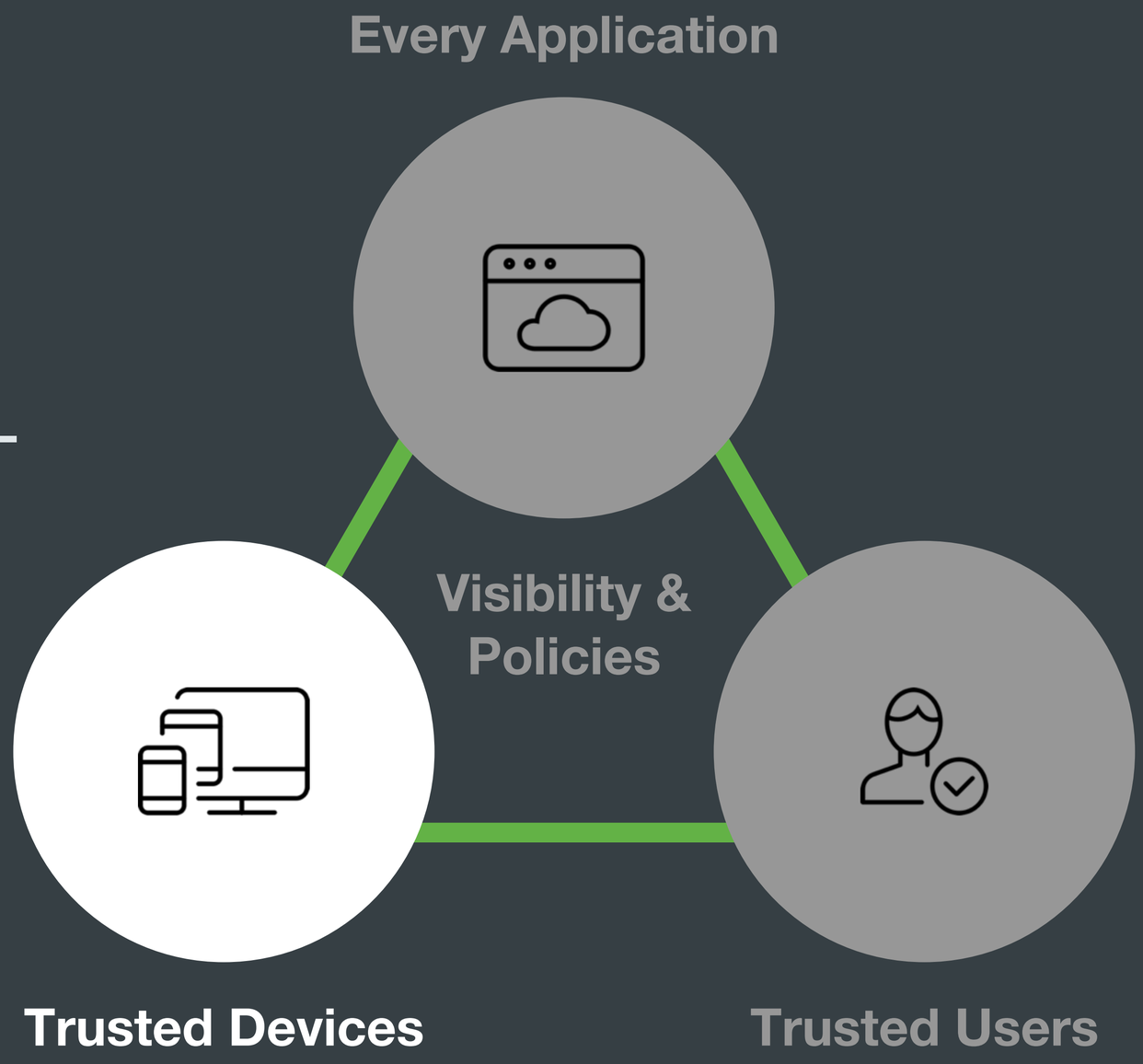
Attackers must compromise:

- Username
- Password
- 2nd auth factor

- Trusted device

Trusted Devices:

- ➔Allow only known devices
- ➔Distinguish user- vs. corp-managed
- ➔Enforce device hygiene
- ➔Regardless of location
- ➔On every access
- ➔For every application



Protect Sensitive Apps From Risky Devices

Example: Devices with Poor Security Hygiene

BYOD

Cloud Apps

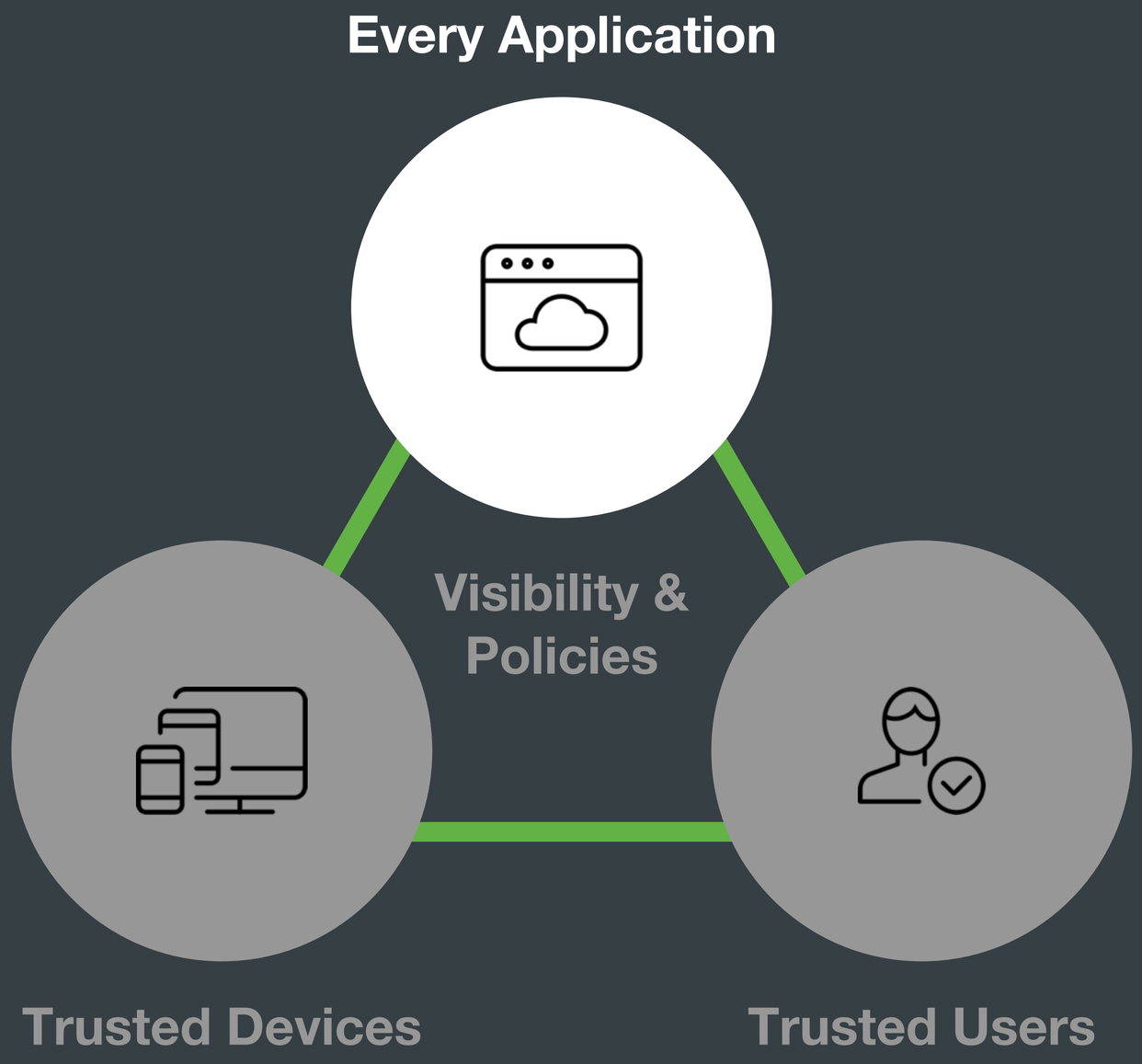
On-Premises Apps

Enforce Device Policy:

- Known device
- Up-to-date software
- Security configuration
- Device risk

Every Application:

- ➔Cloud-based or on-prem
- ➔Regardless of access location
- ➔For every user
- ➔On every access
- ➔For every device



Enforce Policy Based Controls

Get Granular

- Block anonymous networks, out-of-date browsers and plugins, and rooted or jailbroken devices
- Require users to enable screen-lock and use U2F or push authentication
- Ensure all systems are up-to-date

Zero Trust Shopping List

- Asset Inventory.
- User Management.
- Device Management through uniform security policy.
- Defined Repeatable Process.
- User and Entity Behavior Analytics.
- Network Zone Segmentation.

The Authentications Must Flow

Supply Chain Security

SSH

22
TCP
ssh

OpenSSH Version: 5.3

```
SSH-2.0-OpenSSH_5.3
Key type: ssh-rsa
Key: AAAAB3NzaC1yc2EAAAABIwAAAQEAuMYyK4vK
v5sT6ex4XyMvaeXf9SOfgTlLrSdNETbAE5AHoWQbS
5tTBmeKJevMbrSSprPDeISvHIOSBTDh/JK/huvyUN
HiEEUaAWfvSbvTdNbb34XvKDgPb4NiuYtA2Y/isBb+
ssdDmMPBtZ4DBMOL4OBTctt/Sk/BcTwcnCqLCCY8D
Fingerprint: 89:b6:83:c3:a8:eb:...:7f:94:de:7
```

Kex Algorithms:

- diffie-hellman-group-exchange-sha
- diffie-hellman-group-exchange-sha
- diffie-hellman-group14-sha1
- diffie-hellman-group1-sha1

Server Host Key Algorithms:

- ssh-rsa
- ssh-dss

Encryption Algorithms:

- aes128-ctr
- aes192-ctr
- aes256-ctr
- arcfour256
- arcfour128
- aes128-cbc
- 3des-cbc
- blowfish-cbc
- cast128-cbc
- aes192-cbc
- aes256-cbc
- arcfour
- rijndael-

MAC Algorithms:

- hmac-md5
- hmac-sha1
- umac-64@openssh.com
- hmac-sha2-256
- hmac-sha2-512
- hmac-ripemd160
- hmac-ripemd160@openssh.com
- hmac-sha1-96
- hmac-md5-96

⚠ Vulnerabilities

Note: the device may not be impacted by all of these issues. The vulnerabilities are implied based on the software and version.

CVE-2011-5000	The ssh_gssapi_parse_ename function in gss-serv.c in OpenSSH 5.8 and earlier, when gssapi-with-mic authentication is enabled, allows remote authenticated users to cause a denial of service (memory consumption) via a large value in a certain length field. NOTE: there may be limited scenarios in which this issue is relevant.
CVE-2016-10708	sshd in OpenSSH before 7.4 allows remote attackers to cause a denial of service (NULL pointer dereference and daemon crash) via an out-of-sequence NEWKEYS message, as demonstrated by Honggfuzz, related to kex.c and packet.c.
CVE-2014-1692	The hash_buffer function in schnorr.c in OpenSSH through 6.4, when Makefile.inc is modified to enable the J-PAKE protocol, does not initialize certain data structures, which might allow remote attackers to cause a denial of service (memory corruption) or have unspecified other impact via vectors that trigger an error condition.
	...ion of OpenSSH through 6.1 enforces a fixed time limit between establishing a ...ompleting a login, which makes it easier for remote attackers to cause a denial ...n-slot exhaustion) by periodically making many new TCP connections.
	...ction in sftp-server.c in OpenSSH before 7.6 does not properly prevent write ...y mode, which allows attackers to create zero-length files.
CVE-2010-4478	OpenSSH 5.6 and earlier, when J-PAKE is enabled, does not properly validate the public parameters in the J-PAKE protocol, which allows remote attackers to bypass the need for knowledge of the shared secret, and successfully authenticate, by sending crafted values in each round of the protocol, a related issue to CVE-2010-4252.
CVE-2016-0777	The resend_bytes function in roaming_common.c in the client in OpenSSH 5.x, 6.x, and 7.x before 7.1p2 allows remote servers to obtain sensitive information from process memory by requesting transmission of an entire buffer, as demonstrated by reading a private key.
CVE-2011-4327	ssh-keysign.c in ssh-keysign in OpenSSH before 5.8p2 on certain platforms executes ssh-rand-helper with unintended open file descriptors, which allows local users to obtain sensitive key information via the ptrace system call.
CVE-2010-4755	The (1) remote_glob function in sftp-glob.c and the (2) process_put function in sftp.c in OpenSSH 5.8 and earlier, as used in FreeBSD 7.3 and 8.1, NetBSD 5.0.2, OpenBSD 4.7, and other products, allow remote authenticated users to cause a denial of service (CPU and memory consumption) via crafted glob expressions that do not match any pathnames, as demonstrated by glob expressions in SSH_FXP_STAT requests to an sftp daemon, a different vulnerability than CVE-2010-2632.

8,612,794

DUO


ENCRYPT
ALL THE THINGS!

A Better Way Forward

WebAuthN


Biometric Authentication Ecosystem
Hardware
Platform
Open
Standards


LET ME EXPLAIN
NO THERE IS TOO MUCH. LET ME SUM UP.

Zero Trust True-isms

- Assume the network is hostile
- Establish a trust engine
- Reduce threat surface
- Continuously validate for authorization
- KISS

Zero Trust Summary

- Build an asset inventory.
- Get a solid hold on user management.
- What’s on your network?
- Defined Repeatable Process
- User and Entity Behavior Analytics.
- Network Zone Segmentation.

The Sword Is Dissolving

No Need For The Holy Hand Grenade

Thanks For Listening!

Thanks!

gattaca@duo.com
@gattaca
www.duo.com