SPICE WORLD 2019

Learning from Failure:
Tales of Incident Response Gone Wrong

SPTCEWORLD2019

Nick Leghorn

Manager of Security Engineering, Indeed.com

SPICEWORLD2019

DISCLAIMER

None of the incidents depicted within this presentation occurred at Indeed.

Names, specific details, and other facts have been anonymized to protect customer confidentiality.

Don’t fire people the first time they make a big mistake.

They just learned a very expensive lesson, and you paid for that education.

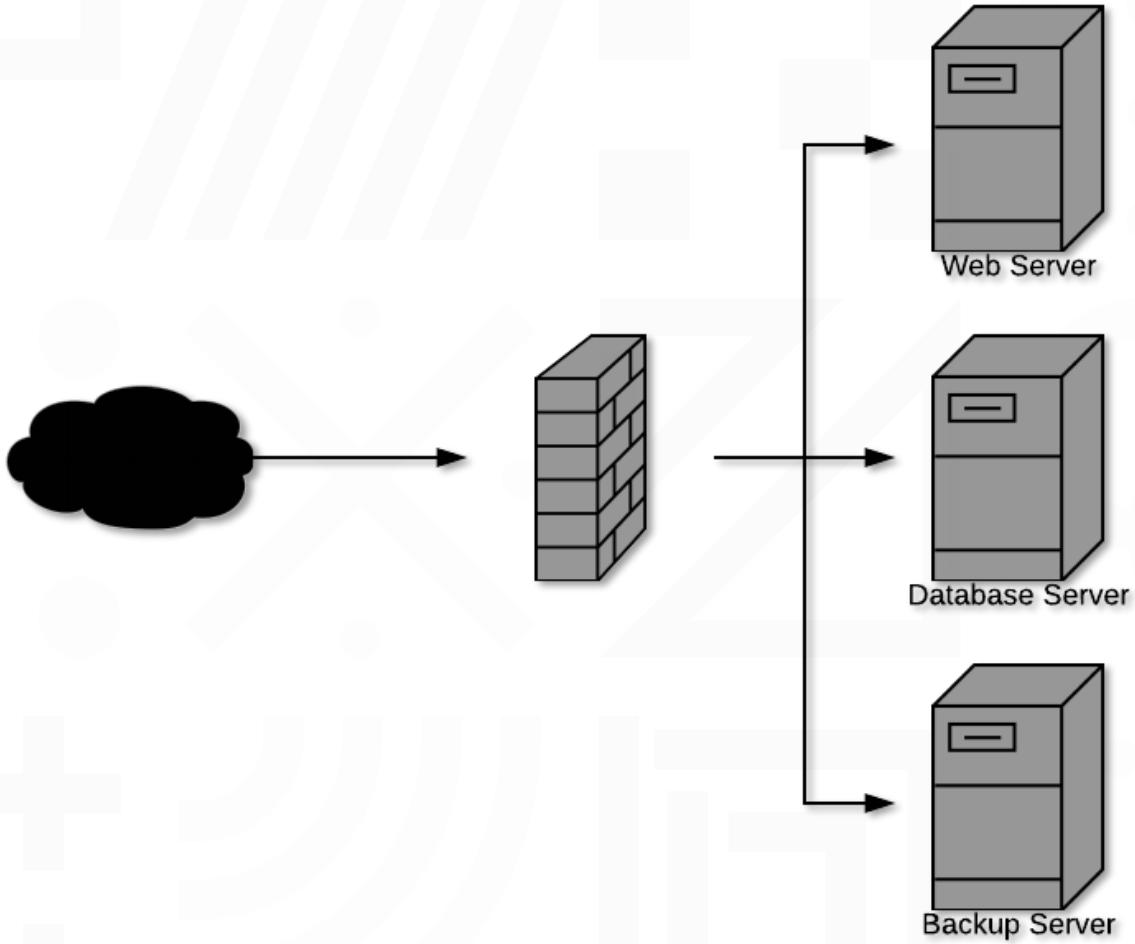

Detection
Containment
Remediation

Chapter One:

When You Assume...

SPICEWORLD2019

Customer A's Setup



Firewall Configuration

```
Access-list 101 extended permit tcp any host [web server] eq www
Access-list 101 extended permit tcp any host [web server] eq https
```

```
Access-list 101 extended permit tcp [Office IP] any eq 3389
Access-list 101 extended permit udp [Office IP] any eq 3389
Access-list 101 extended permit tcp [Office IP] any eq 3306
```

Access-list 101 extended deny ip any any

Firewall Configuration

```
Access-list 101 extended permit tcp any host [web server] eq www
Access-list 101 extended permit tcp any host [web server] eq https
```

```
Access-list 101 extended permit tcp [Office IP] any eq 3389
Access-list 101 extended permit udp [Office IP] any eq 3389
Access-list 101 extended permit tcp [Office IP] any eq 3306
```

```
Access-list 101 extended permit ip any any
```

Friday, 4:45 PM

I note the firewall security issue in a ticket.

Customer closes ticket.

Monday, 7:45 AM

Website reported offline. Users notified.

Monday, 8:00 AM
(15 minutes since first alert)

DB Admins arrive at work. Database server is unresponsive.

Monday, 8:45 AM
(1 hour since first alert)

Admins attempt to RDP into web server.


MAKTUB LOCKER
WARNING!
Your personal files are encrypted!
11:58:26
Your documents, photos, databases and other important files have been encrypted with strongest encryption and unique key, generated for this computer. Private decryption key is stored on a secret Internet server and nobody can decrypt your files until you pay and obtain the private key. The server will eliminate the key after a time period specified in this window.
Open http://maktubuyatq4rfyo.onion.link
or http://maktubuyatq4rfyo.torstorm.org
or http://maktubuyatq4rfyo.tor2web.org

Monday, 9:15 AM
(1:30 since first alert)

Server team contacted, asked to initiate restore process.

Server team informs that customer opted out of tape backups, only has file backups on network server.

Monday, 10:00 AM
(2:15 since first alert)

Networked backup server unresponsive.

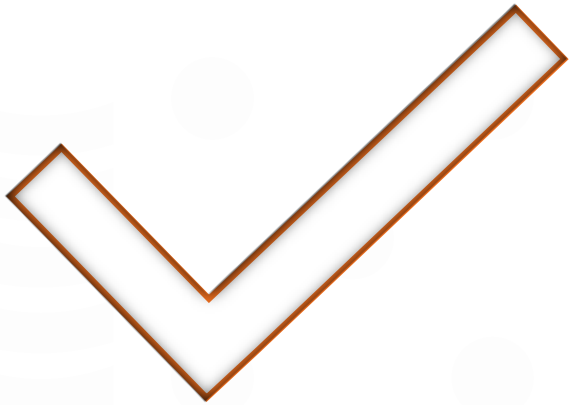
Server team tries to log in, finds the same ransomware banner.

Monday, 10:30 AM
(2:45 since first alert)

Firewall logs show repeated RDP session connections from unknown IPs going on for weeks.

System admins admit the same password is used on all servers.

There is no internal segmentation.



Check box security:

"Our site is secure because we bought a firewall"

Monday, 2:00 PM

Meeting with company.

Discussion about paying the ransom.

Decide to format and reinstall with dev environment as source.

Tuesday, 3:00 PM

New web server and database server provisioned,
still need data to be restored.

Wednesday, 10:00 AM

Database files not encrypted and were salvaged from the compromised backup server.

Only files < 1 GB were impacted.

Thursday, 9:00 AM

Website and database restored to normal operation.

Total downtime: three days.

Lessons Learned

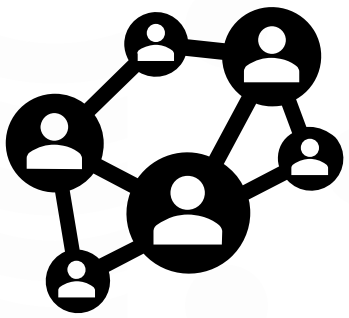
1. Don't assume that you are secure just because you bought a device or service
2. VPN: Annoying for executives, but necessary for security
3. Admin credential re-use is a no-no
4. Proper network segmentation is always a good idea
5. Logging & monitoring is critical, and IDS would have been super helpful
6. The customer got lucky - it could have been much worse

Chapter Two:

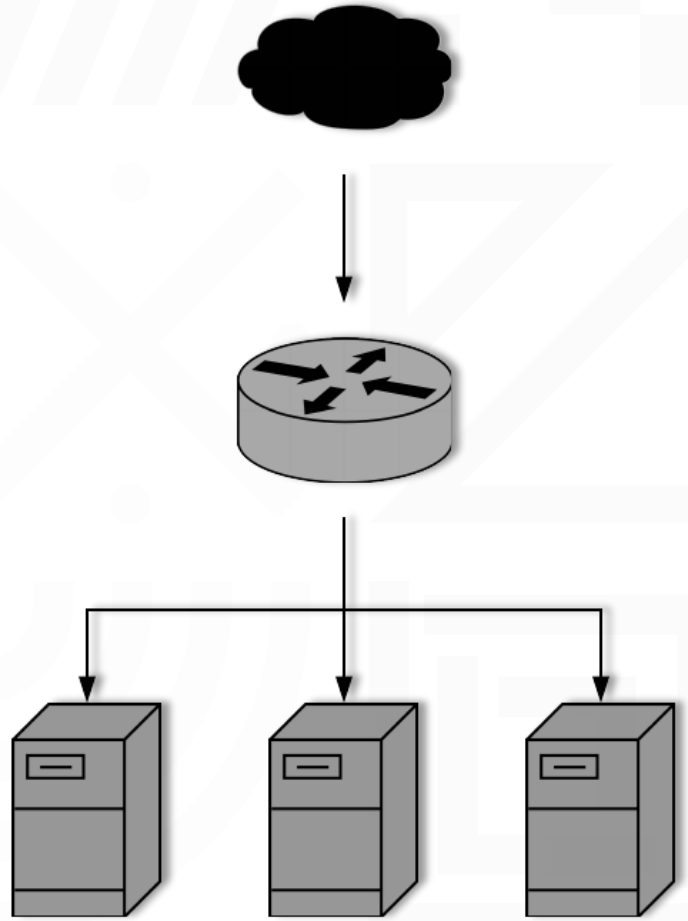
Documentation Only Works if People Know About It

SPICEWORLD2019

DDoS: Distributed Denial of Service



Typical network configuration can impact other customers



DDoS Standard Operating Procedure

1. Identify the targeted customer

2. Contact the customer and alert them of the situation

3. Recommend DDoS protection service ($$$$)

4. If (customer doesn't respond OR declines protection)
   AND (DDoS impacts other customers)
   THEN (null route IP)

Ticketing systems are not always your friend

[Account#] - [Customer Name] | [Ticket Title]

Account Info
Devices
Credentials
Websites
IP Information
Tickets

Account Notes

On [Date] at [time] [customer] wrote…

OMG ALL THE THINGS ARE BROKEN PLZ FIX!

Attack +0:15m

Monitoring picks up high interface usage, triggers alert.

Attack +0:20m

SSH into firewall, note large number of connections but firewall is coping well.

Attack +0:30m

One website associated with account is no longer responding, triggers an alert.

Attack +0:45m

Identify primary contact on the account.

Attempt to contact, goes to voicemail. Update ticket with information.

Attack +1:30h

No response from primary contact.

Customers in adjacent cabinets start to complain about website speed.

DDoS Standard Operating Procedure

1. Identify the targeted customer
2. Contact the customer and alert them of the situation
3. Recommend DDoS protection service ($$$$)
4. If (**customer doesn't respond** OR declines protection)
   AND (**DDoS impacts other customers**)
   THEN (null route IP)

Attack +1:45h

Attacked customer's IP null routed at the edge of the network.

Attack stops, traffic returns to normal.

Attack +6:15h

Customer's primary contact calls and asks why DDoS protection wasn't applied.

Customer's account manager confirms that they bought the protection plan.

Attack +6:30h

Null route removed.

DDoS protection applied.

Attack no longer impacting customers.

Attack + 24 hours: Media reports downtime

The computer networks for [service] were off line for most of [date], possibly because of an attack by a group of hackers.

The hackers claimed responsibility on Twitter for the shutdown, which essentially rendered the [service] unusable, but the company denied the source of its problems. A spokesman on Friday said the network was back up, but he declined to comment on the cause. Press officials did not immediately respond to requests for comment.

- NY Times

Ticketing systems are not always your friend

[Account#] - [Customer Name] | [Ticket Title]

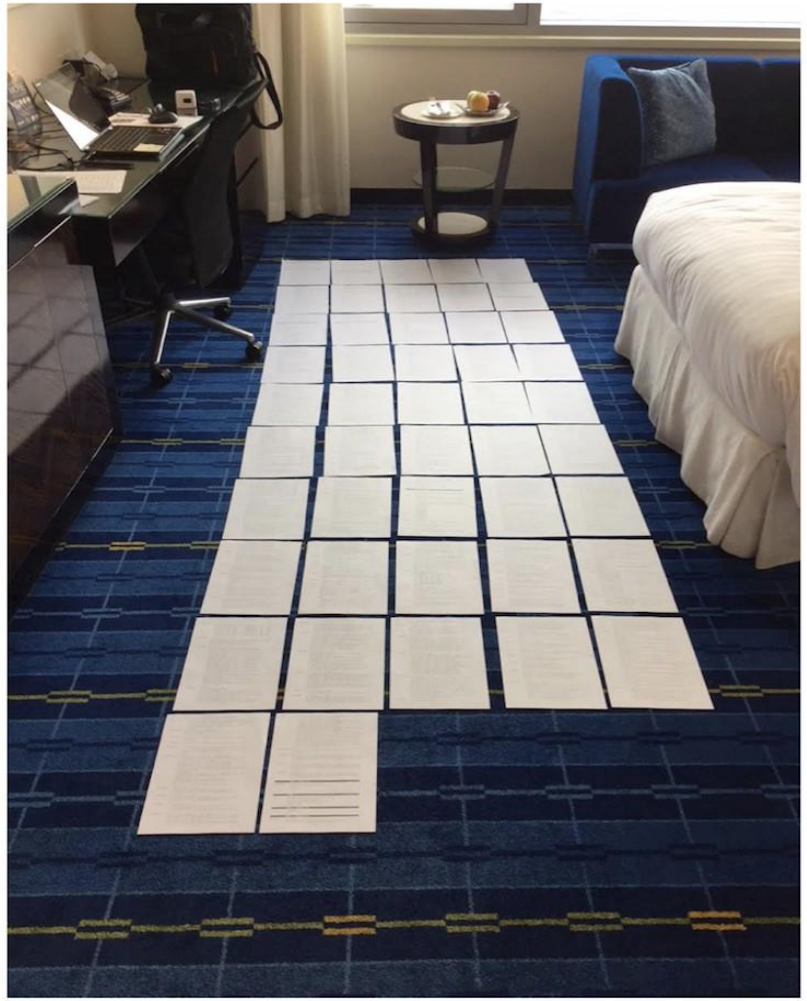
Account Info
Devices
Credentials
Websites
IP Information
Tickets

Account Notes

On [Date] at [time] [customer] wrote…

OMG ALL THE THINGS ARE BROKEN PLZ FIX!

Ticketing systems are not always your friend



Lessons Learned

1. Incident response documentation needs to be CLEAR and CONSISTENT
    a. Large blocks of text that an IR tech needs to parse during critical situations is not useful
    b. Use bullet points, bold consistent headings, and diagrams where possible
2. Documentation should be constantly updated
3. Frequent table top exercises help reduce stress and promote consistent and correct actions

Chapter Three:

Can't Stop the Signal.

Or the Noise.

SPICEWORLD2019

Protecting the Customer's Website

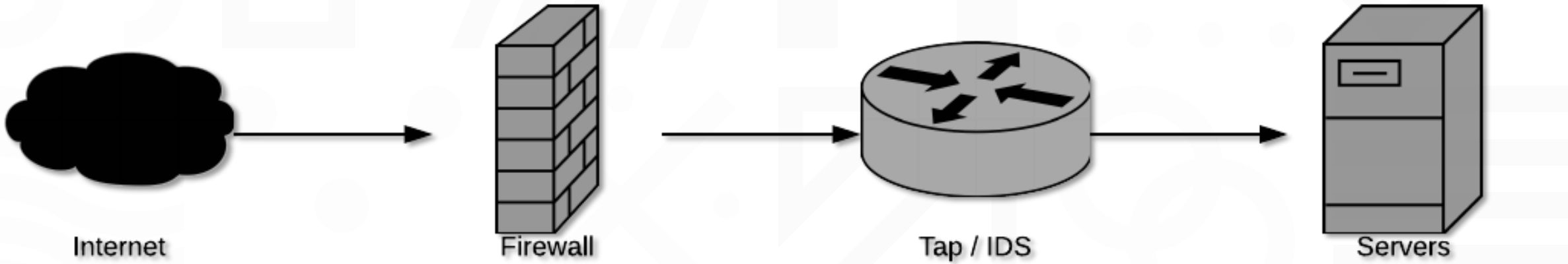
Customer is a vendor who runs an online community for children (think message board) to interact with their client's brand ambassadors.

- Customer's client demands security features as part of the contract
  - Firewall
  - IDS
  - Website monitoring

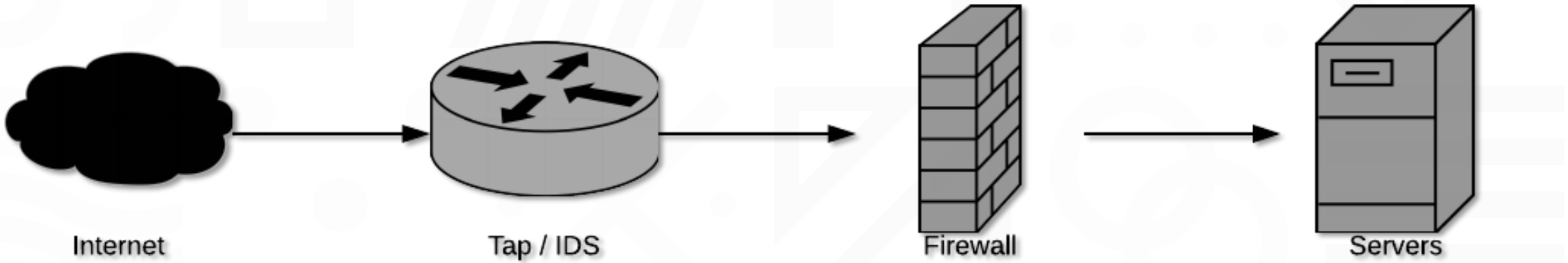
- Customer decides they want a non-standard deployment

IDS: Intrusion Detection Sensor

Normal IDS Deployment



Customer's IDS Deployment



Day One
IDS Uptime: 5 minutes

IDS detects an attack and opens a ticket.

We investigate, report to customer.

Customer investigates, reports all clear.

We close ticket.

Day One
IDS Uptime: 30 minutes

IDS detects an attack and opens a ticket.

We investigate, report to customer.

Customer investigates, reports all clear.

We close ticket.

Day One
IDS Uptime: 55 minutes

IDS detects an attack and opens a ticket.

We investigate, report to customer.

Customer investigates, reports all clear.

We close ticket.

Day One
IDS Uptime: 2 hours

IDS detects an attack and opens a ticket.

We investigate, report to customer.

Customer **DOES NOT** report all clear.

Customer closes the ticket.

Eventually...

IDS detects an attack and opens a ticket.

We investigate, report to customer.

Customer does not respond.

48 hours later we close the ticket.


MUCH, MUCH,
MUCH LATER

Customer Ticket #1

IDS detects an attack and opens a ticket.

We investigate, report to customer.

Customer does not respond.

48 hours later we close the ticket.

Customer Ticket #2

Customer reports that their client's website has been hacked.

Customer Ticket #2

Customer demands to know why we didn't stop it.

We reference ticket #1.

Root cause: no rate limiting / brute force detection in login logic.

Customer refuses to take responsibility, tells client that the hack was host's fault and refuses to fix code.

This happens again.

And again.

Five more times.

Client gets report of what the actual root cause was.

Client fires customer.

Customer goes out of business.

Lessons Learned

1. All alerts should be actionable.

2. "Zero blame retrospectives" are helpful
    a. The customer was more concerned with being "blame free" in the eyes of their client than fixing the problem

3. Compliance for compliance sake doesn't work. Security needs to be correctly implemented to block actual attackers.

Key Takeaways

- "Check the box" security isn't security.
  - Use security experts to design your defense in depth strategy.
  - Design actionable alerts and investigate them.
  - Encourage reporting of insecure situations.
- Have an incident response plan.
  - Document the plan.
  - Hold table top exercises.
  - Make sure everyone is on the same page.
- Every mistake is a learning opportunity.
  - Do a zero blame post-mortem for every incident.
  - Don't fire someone for their *first* big mistake.

Instagram / Twitter

foghorn@NickLeghorn.com

GitHub

Website

Email

SPICEWORLD2019

Thank you.

SPTCEWORLD2019