PENTEST SECRETS

Breaking The Unbreakable Enterprise Security

AUTHOR - SAGAR BANSAL
CO-AUTHOR - AJAY BARALA

ISBN: 979-8-67494-954-1

1st Edition: August 2020

~ SAGAR BANSAL

ACKNOWLEDGEMENTS

Several Individuals have made this book possible.

I would like to first express my sincere thanks to **Almighty God**, who bestowed upon me thought and wisdom to write this book.

I thank all my **Family & Friends** for their always coming encouragement, guidance, inspiration, love and support, which made me able to achieve everything that I have got in my life.

A Sincere thanks to the whole **Sagar Bansal Digital Team** for their marvellous efforts and dedicated hard work.

Marco Essomba and BlockAPT Team for all the technical insights on enterprise security devices.

Ajay Prakash and USkill Team for strategic support with all the connections.

PJ Professional Services Team for giving us support with their testing infrastructure.

Kenneth Klutse and ARKS Data Solutions Team for all helping with Risk & Compliance issues.

Atsunyo Solutions Team and Kairos Vision Consult Team for helping with NDA issues.

Safinah Murad for converting our screenshots, photographs, thoughts and ideas into amazing illustrations.

Avinash Yadav and Jissy Davis for regularly reviewing the book and providing great suggestions.

At last, a special thanks to **My Students** all over the world, who keep admiring my work and give me all the energy and enthusiasm to create more awesome content.

~ SAGAR BANSAL

TABLE OF CONTENT

BEFORE YOU READ

When I sent this book to some of my students for review before publishing, They All assumed that I would show them terminal screens, tools, techniques and all that stuff which you usually would expect from a book about pentesting.

Technology changes every day. Back in the 2000s if you were able to hack Windows XP, you would be working in Microsoft at an executive position. In today's world, even a school kid can use EternalBlue and crack it open.

You are reading this book because you want to reach somewhere in your life. This book is not a guide. No one can tell you precisely what steps you have to take. There is just no system which you can follow as it is.

Instead of a How-To Guide, my intent with this book is to give you enough exposure to the industry's deep secrets and give you a mindset to be the person you want to be.

It's the essence which is missing in your life.

SECRET #1

TO DO A PENTEST, YOU NEED A TEAM

Majority of people believe that they can be the greatest pentesters of all. This statement goes void at its existence itself because pentesting is not a one-person show.

It's a team-based service model. Whether you are working as a technician in a company or doing a double-blind pentest as a service provider, you never work alone.

There are at least 3 to 4 people in any pentesting team. Everyone has his or her expertise areas. As you can imagine, if there is a team, you need a leader.

Yes, you will face problems, there will be tough situations and complex challenges that you will have to overcome, and your leader will show you the path. He makes the hard calls. He pulls all the strings.

I manage this global team of 28 people. Let me introduce you to some key players of my team who did this fantastic pentest...

Yes, we are talking about my journey of making over a quarter-million dollars in the double-blind pentesting industry in just 3 months.

I feel safe to assume you already know who Sagar Bansal is.

Yes, The Author of this Book, A Celebrity Expert, Mentor to over 35K+ CISO's leading the World's Infosec Economy. It's Me!

Professionally I hold a Certified CISO, LPT Master, ECSA, CHFI, CEH, CEI as my EC-Council Certifications. I am also one of the EC-Council Global Advisory Board Members - Ethical Hacking For India, Middle East & Africa Region.

My Core Expertise lies in Project Management which comes from my Executive Degree from Indian Institute of Management, GRC which comes from CCISO CBK, and Stakeholder Management which comes from extensive experience with multiple Fortune 500 Companies.

My role in this team?
Well, I am doing everything!

I will be handling the client relations, deals, project, processes and most importantly - all other team members.

Ajay is the most senior guy we have in our team. He is a retired X-INDIAN-NAVY Lt. Commander who has strong expertise in GRC & Digital Forensics. Due to his combat experience in defence, He is excellent at planning the strategy, selecting the best methodology and working people hell out of their calibre.

Professionally Ajay holds over 32 certifications including CISSP, GCIH, OSCP, CHFI, CSAP, CEH, OSFTC, OPSE, Security+, CCNA, ISO 27001 LA to name a few. Did I mention his Multiple Master Degrees?

You may be wondering that this man has immense knowledge, so what exactly is his role in this pentest?

To get the best out of all this experience and talent, I decided to include Ajay as much as possible. He was with me in meetings with the client, He was

there when we got access to systems, Heck he was there while writing reports and delivering presentations as well.

He was there in every single phase of this project, and that's the reason why Ajay is the co-author of this book.

I don't think Jitendra needs any introduction. He is a celebrity author himself with over 6 International Best Selling Programs.

He has been working with me for the last five years on Confidential Government Projects in US, UK, Australia, India and Dubai.

Throughout his career, he has reported nasty bugs to most of the Fortune 500 companies, including Facebook, Google, Medium and many more.

Jitendra holds his Master's degree in Computer Applications with a major focus on Web Application & Mobile API Testing.

For this project, He was responsible for managing other team members and heading the Web App Related Tests.

Initially, Mandeep was my student, but he worked hard and paved his way into enterprise security.

He holds his Master's Degree in Information Security from MIT, Melbourne and today he is ranked in the Top 100 External Enterprise Security Infrastructure SMEs in Australia.

He is also the co-author of my Pentest Career Blueprint program, and he assists me in all my live sessions related to Infrastructure Security.

Naturally, he handles Infrastructure Testing works in my Global Team under Paul's supervision.

Wait for what?

Ms X is an essential member of my team, but she is working on some national security project, and we cannot reveal her real name.

She is our Senior Exploit Writer. She holds MSc and GXPN as some of her credentials. Again, I wouldn't be giving the full list to ensure protection against any aggregation attack. (we know our stuff)

Ms X is not permanent in my global team. Instead, her charges are so high that we could only afford her for 80 hours of service as per the project's budget.

Paul is one of the Top Advisors to Federal CISO's, CEO's and CIO's providing insights and solutions as a Subject Matter Expert in the United States.

He is a well-known author on FISMA, FedRamp, GRC and Network Pentesting.

Paul holds an MBA as well as an MS. in CyberSecurity. Apart from CISSP, PMP, PSM1, Security+, CEI, CEH, he holds over 20+ vendor-specific certifications.

He is the one who was handling most of the network pentesting work and also making sure we stay in compliance and avoid any legal complications. Well, I want to continue this list, but then half of this book will just be introduction pages.

If you are a potential client for my business, I would love to give you the full company profile if we seem to be a good fit.

But till then, let's keep going further for now.

As I said, for any penetration test, you need a team on which you can depend on. If you are reading it with a mindset of a one-man army, you will not understand anything.

So let me ask you this question…

Can you open your mind to the extent that everything you have seen in the industry till now, everything you learnt in courses and programs, everything you have been told about… seems wrong and fake?

Will you be flexible enough to accept the harsh reality of this industry which is somewhere hidden in the glory of false demand?

Are you ready to accept the truth and learn from this book?

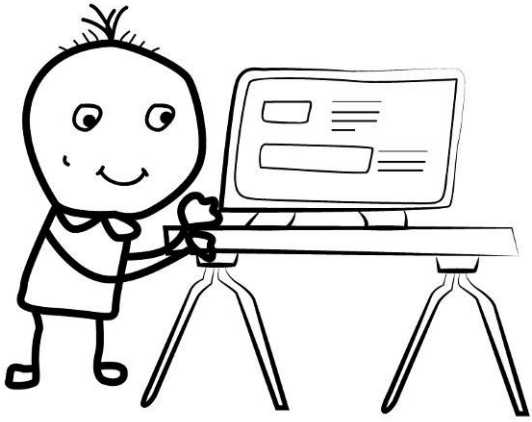
If the answer is yes, and indeed yes… Then go ahead and read the next chapter. I promise this book will change your life, and you will never feel the same way again as you do now.

See you in the next chapter!

SECRET #2

EVERYTHING YOU STUDIED IS WRONG

You might have seen some movies, read some books, attended some workshops or even done some certification exam in this field. But you know nothing about Enterprise Security.



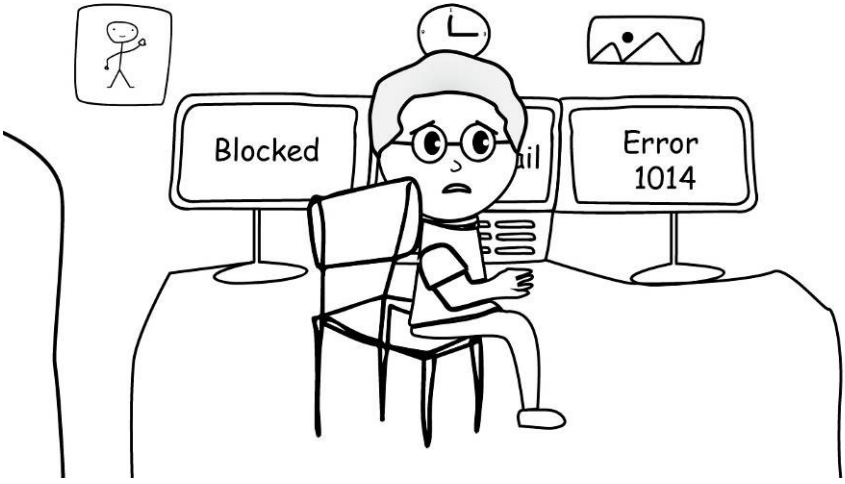
For me, you are nothing more than a sweet kid who has learnt some basics of red teaming and is now trying to dive into the world of pentesting.

No seriously, I mean every single word in my statement!

An Enterprise-Grade Security involves multiple defence mechanisms which are often put into a Full Stack Security Model.

When you start doing initial reconnaissance, you may ping the website, perform zone transfers, scan it with technology profilers and do all that standard stuff.

I did the same too, and after some time, this was me!



When I attempted a simple ping, I got an Edge Server of Cloudflare, which was useless for us. This IP is not of an original server.

```
C:\Users\sagar>ping target.gov

Pinging target.gov [104.24.121.220] with 32 bytes of data:
Reply from 104.24.121.220: bytes=32 time=395ms TTL=56
Reply from 104.24.121.220: bytes=32 time=413ms TTL=56
Reply from 104.24.121.220: bytes=32 time=427ms TTL=56
Reply from 104.24.121.220: bytes=32 time=443ms TTL=56

Ping statistics for 104.24.121.220:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 395ms, Maximum = 443ms, Average = 419ms
```

When we attempted a Zone Transfer, Our IP got straight blacklisted.

```
root@sagarbansal:~# dig +short ns target.gov
liz.ns.cloudflare.com.
skip.ns.cloudflare.com.
root@sagarbansal:~# dig axfr @liz.ns.cloudflare.com target.gov

; <<>> DiG 9.11.5-P4-5.1+b1-Debian <<>> axfr @liz.ns.cloudflare.com target.gov
; (2 servers found)
;; global options: +cmd
; Transfer failed.
root@sagarbansal:~# dig axfr @skip.ns.cloudflare.com target.gov

; <<>> DiG 9.11.5-P4-5.1+b1-Debian <<>> axfr @skip.ns.cloudflare.com target.gov
; (2 servers found)
;; global options: +cmd
; Transfer failed.
```

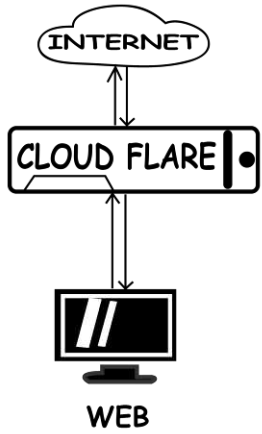
When we tried using a technology profiler like Builtwith and WhatWeb, We got zero results.

```
root@sagarbansal:~# whatweb target.gov
http://target.gov [403 Forbidden] CloudFlare, Cookies[__cfduid], Country[UNITED
STATES][US], HTML5, HTTPServer[cloudflare], HttpOnly[__cfduid], IP[104.24.121.220],
Script[text/javascript], Title[Home | target.gov | Cloudflare],
 UncommonHeaders[cf-request-id,cf-cache-status,cf-ray],
 X-Frame-Options[SAMEORIGIN], X-UA-Compatible[IE=Edge]
```

It's all blank.

For those of you who don't know what Cloudflare is…

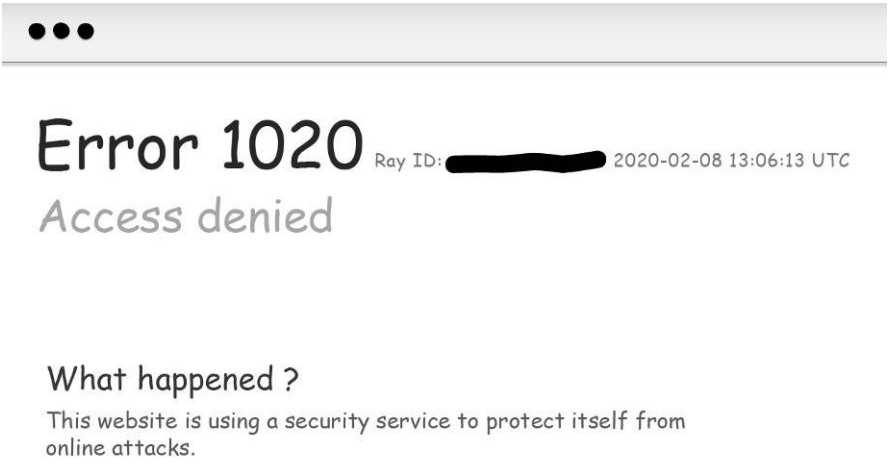
It is basically a Reverse Proxy which comes with hundreds of web attack protection.



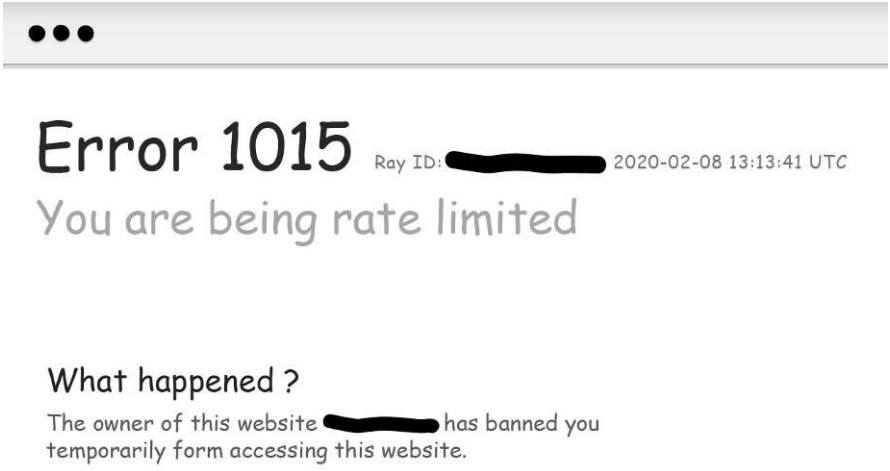
Enterprise Account of Cloudflare has pre-configured rules against OWASP Top 10 attacks. Common attack patterns won't work here.

Additionally, the security team of this company has done a fantastic job because they are throttling the user requests in various ways.

If you open /login or /admin pages, you will get Error 1020 which is a Blocking Error by Cloudflare Firewall.



If you open more than 15 pages of this website in a minute, you will get Error 1015 which is a Rate Limiting Response by Cloudflare Firewall.



As you can imagine, if it is rate-limiting a human itself, what it will do with automated scanners and tools.

It's just impossible to use things like Nessus, Qualys, Vega, LFI, Saint or any other industry-grade tool that you might have ever heard of.

If you try to use a directory bursting attack, you will get blocked by Cloudflare, and it will show every response as 403

```
root@sagarbansal:~# dirb https://target.gov

-----------------
DIRB v2.22
By The Dark Raver
-----------------

START_TIME: Thu Feb  9 08:35:37 2020
URL_BASE: https://target.gov/
WORDLIST_FILES: /usr/share/dirb/wordlists/common.txt

-----------------

GENERATED WORDS: 4612

---- Scanning URL: https://target.gov/ ----

(!) FATAL: Too many errors connecting to host
   (Possible cause: OPERATION TIMEOUT)

-----------------
END_TIME: Thu Feb  9 08:38:27 2020
DOWNLOADED: 24 - FOUND: 0
```

Same happens with tools like DirBuster as well. It seems only selected user-agents are allowed to access the website, and there is a perfect blend of custom security rules.

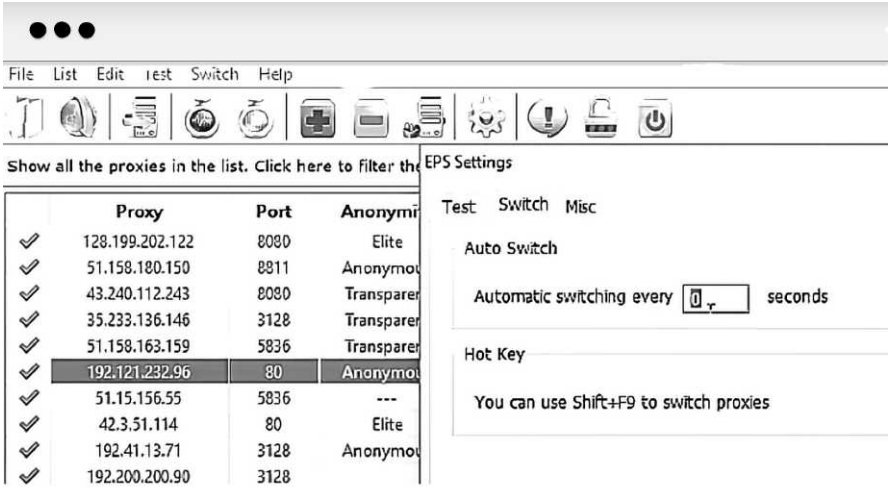
We have to map these rules. There is just no other way. It's a trial and error game now. The only problem is time. We only have 3 months for this project, and every day is costing us more than 1% of our time.

I asked my team to create a custom script which rotates between user-agents while sending any request. Our exploit writer coded a script based on Nginx Transparent Proxy. You can see that we were replacing **$agent** variable with random agents from a list.

```
http {
   server {
      location / {
         proxy_pass          http://$host$request_uri;
         proxy_set_header      "User-Agent" "$agent";
         proxy_connect_timeout  60;
         proxy_send_timeout     60;
         proxy_read_timeout     60;
      }
   }
}
```

Imagine how hard it is when your IP is getting banned every minute. It's frustrating!

Hence, We used a rotating IP Proxy called Elite Proxy Switcher, which helped us to change IP Address every 60 seconds. (I am not endorsing the product but it's super awesome and only costs around $30)



My team kept mapping stuff slowly, finally, after getting banned with over 300+ IP Addresses, wasting over a whole day.

We now know that we can send unlimited requests using the Google Bot Mobile User-Agent as it seems like they don't want to ban Google for SEO purposes.

Mozilla/5.0 (Linux; Android 5.0; SM-G920A) AppleWebKit (KHTML, like Gecko) Chrome Mobile Safari (compatible; AdsBot-Google-Mobile; +http://www.google.com/mobile/adsbot.html)

But hey…
This is not at all enough. We need to find some serious information on this website.

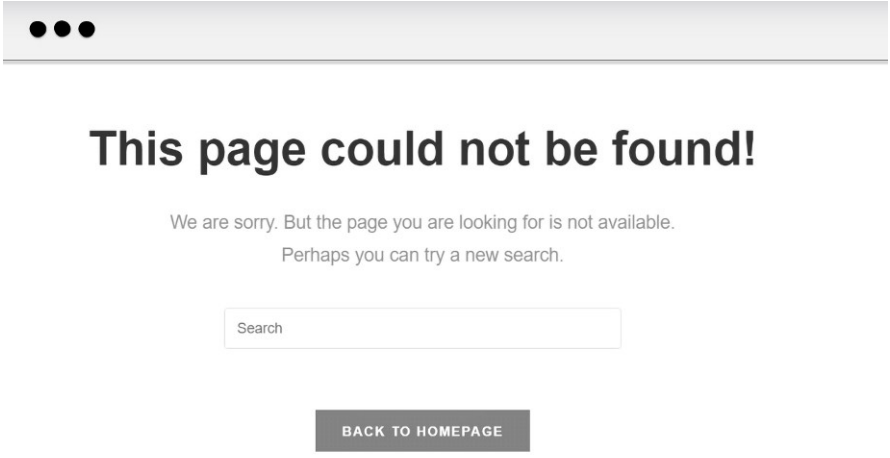
The problem is, we cannot try any direct attack like an SQL Injection, Cloudflare will block everything. Further, there would definitely be a Web Application

Firewall on the website also, and Don't forget other Endpoint Security Devices like DLP, AV, IDS etc., etc.

And Even if we could, We don't really have a direct input field like a user login page. Everything is giving Error 1020 - Access Denied even to this custom new user agent of ours.

My team kept analyzing the source of these pages, Intercepted requests in BurpSuite and did whatever they could for two more days. We found some interesting things about the website.

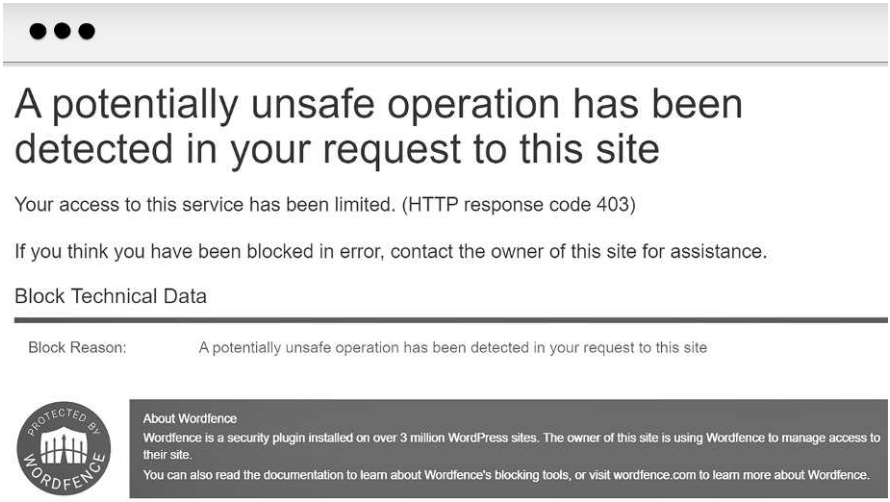
It is running on WordPress. No, we didn't find it written anywhere. It was a pure guess by looking at the 404 page whose design was extremely familiar to Jitendra. He has been testing WordPress for years, and he was damn sure about this. He even said this looks like OceanWP theme of WordPress.



The bad news was that wpscan tool was not working with the custom Google Bot User-Agent. This was to be done manually.

It took us three days to manually map out the whole application by visiting hundreds of URLs.

Sadly even after doing all of this, the website was completely secure, running a WAF called WordFence with no outdated plugins and known vulnerabilities. It looked like an impossible task to get through it, and we didn't really have another clue of where to go next.



It was already the last day of the week, These Cloudflare rules have devastated our performance. Potentially 2 hours of work has taken us 5 days and still nothing beneficial. Everyone's morale was super down, so I decided to leave it for the weekend and took everyone out for dinner.

SECRET #3

SOCIAL ENGINEERING IS HIGHLY TECHNICAL

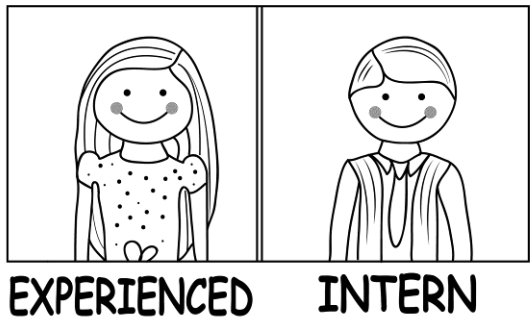
Till now it's absolute that for a pentest which has a limited time of 3 months, it is virtually impossible to break the system using everyday stuff. We just don't have enough time and resources.

We can scan and try different things, but we are changing our IP address every 1 minute. In that one minute time, we can send only 4-5 requests. Even 1-2 sometimes.

It is impossible to guess every single rule they might have put on that Cloudflare. We need those rules. It can only come by compromising an administrator's computer who handles these.

I started looking out for employees which can be potential targets. With Linkedin, it is a matter of minutes to find people working in an organization.

Interestingly, People are so foolish that they put their company name as well as the exact security designation. This is not at all a good thing to do. Maybe say working as Administrator at Confidential. Why do you tell the whole world that Hey! I work as a security admin in this exact organization.



Within 1 hour of research, We had 2 names in front of us. One male and one female. This lady has been working in this organization for the last 4 years. She has an experience of over 10 years in 2 more companies. The gentleman, on the other hand, is doing an Internship. Graduated a few months back and joined this organization this month itself.

Do you think he can be an easy target?

We are targeting a security administrator. These people are smart enough to craft Cloudflare rules that wasted more than a week of our time. It's not going to be this simple!

In Enterprise-Security, you can not just send a phishing e-mail to anyone or just spoof a phone call and expect to get access. All of that looks good in movies.

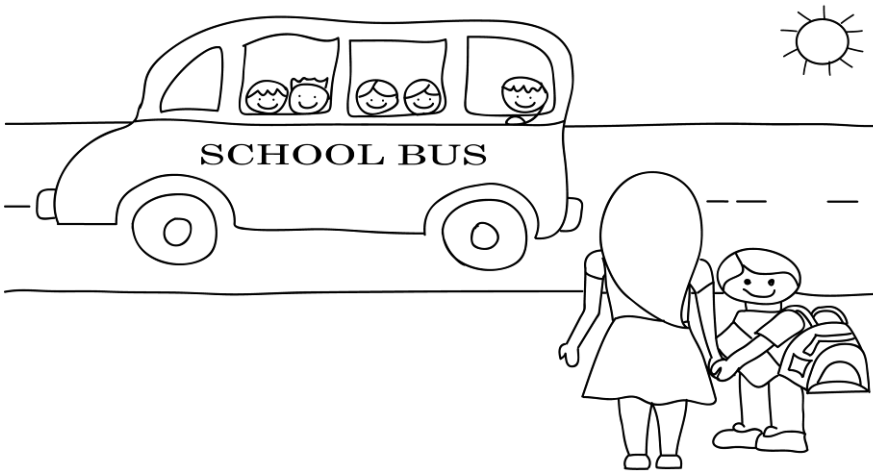
This is our only attempt… I knew we needed more data.
So this is what I e-mailed my team.

Research as much as you can about these people. Try to find out what they like, where they live, where they hang out, their personal goals… Just provide me with as much as data you can within the next 24 hours.

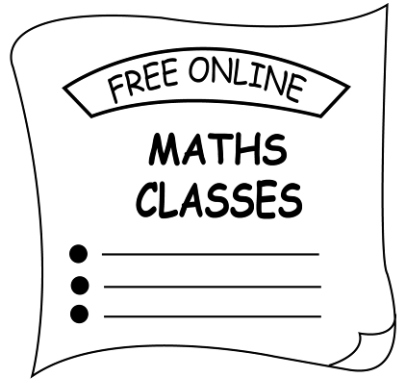
My team spent over 18 hours working on research about the two. Yes, they did use phone calls but not to get access, instead to get information about targets.

I logged into my system at sharp 5:30. I had these sheets created by my team summarizing everything on network diagrams in front of me. It's time to create the plan!

The information which caught my attention was that this hard looking target is actually a married woman who has a kid—a little boy who is in primary school.



This changes everything. My first preference is not that gentleman, but rather this lady now. Parenting is not an easy task… Especially when you do a job in information security.
We used this as our advantage and thought about an advertising flyer for Free Online Maths Classes. (Everyone is weak in this, lol)

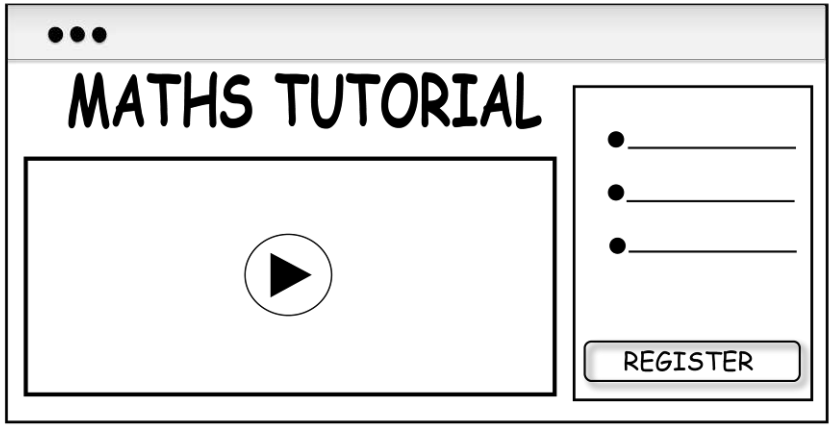


How hard is it to be in a different country and send an advertisement flyer?

Can you not just courier it, right? It will have that international shipping label and all!

So you need a local person to do this for you. I went on fiverr.com and found a graphic designer in the same city. I told him to drop this flyer at her house in exchange for $100, and he agreed.

In the backend, I hired a developer from freelancer.com, who created a whole website for us within a day. Simultaneously, I recorded a 3-minute video about the platform, the mission, the quality education and all.



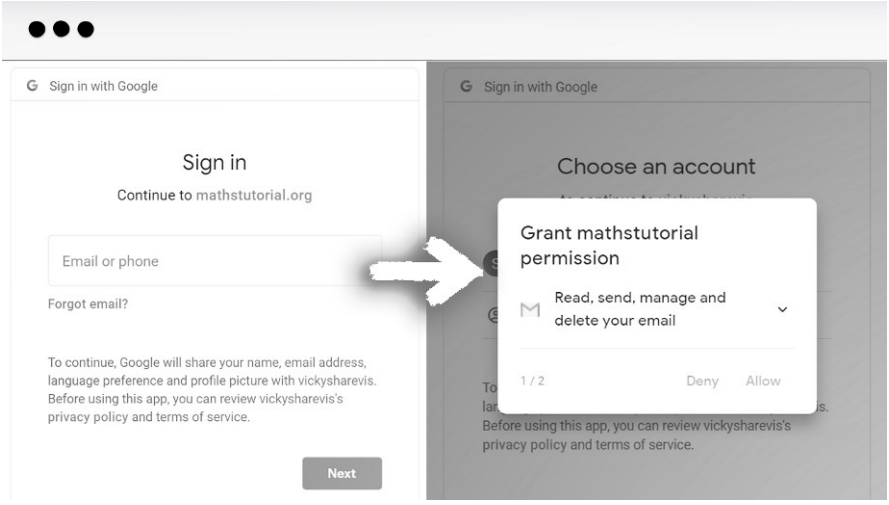
We are pretending to be a new brand called MathsTutorial. Our platform is officially launching in a week, and currently, we are accepting beta customers who are getting free access for a month in return of a testimonial.

This was a perfect pitch for any parent who is super busy for her child's education.

If we use a simple username and password form, She will not reuse any password while creating an account for her child. This would be useless. Hence I asked this developer to create that button as Sign In With Google.

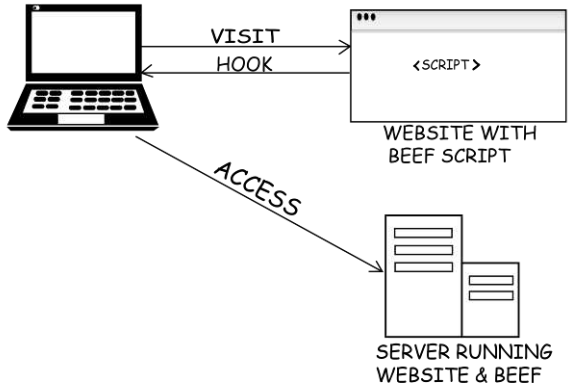
Typically it is based on OAuth 2.0 and OpenID Connect Flow where the user gives permission to the website for fetching profile details from the Google Account.

The trick was that we did not ask for profile details! Instead, we asked for permissions to manage her Gmail using a Restricted Scope. (We already had approval)



This was one part of my attack vector. For the second level of this attack, We installed BeeF on

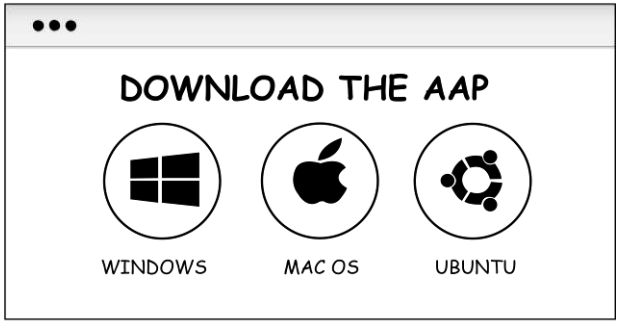
that same web server and put the Payload on our website header. Our BeeF was set on auto exploitation vectors, including Stealing all stored passwords, persistence script, and much more stuff.



Here also a problem was that maybe some DLP or similar Endpoint solution on her laptop will block it and our attack will fail. Hence we restricted the script to only load when a user is logged in. This makes sure that even if we get blocked, We have that e-mail access in the first place.

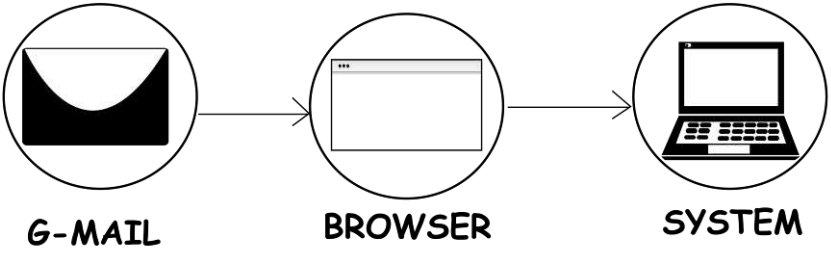
To finish up, I went with another vector by offering a software file to be installed on the computer to access the content.

For this, I hired a software developer from upwork.com, who provided me with a dummy software installer which pops an error in the end.



As I didn't know what operating system she was using, I got a version for each operating system - Windows, macOS & Ubuntu.

Then we injected an obfuscated meterpreter shell which was set to establish a reverse connection to our handler which is listening on the same server of the website in the backend over https.
So as you can imagine, this is a multi-layer attack



1. Stage 1 gives us access to e-mail
2. Stage 2 gives us access to her browser
3. Stage 3 gives us access to her whole system

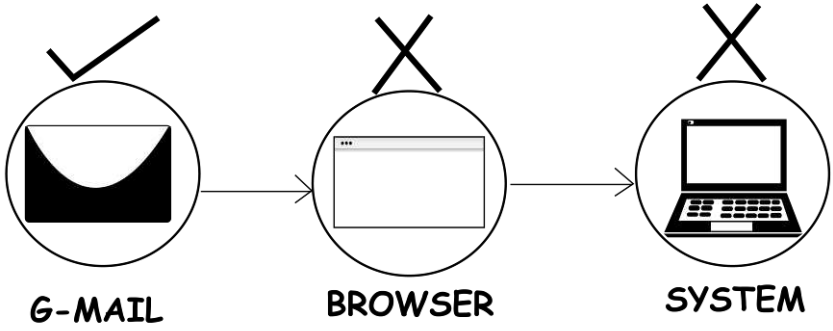
We sent the flyer, and it was just a game of waiting for her to get into this.

We waited for two days… Nothing happened!

It was Sunday morning, My team is enjoying their weekend. But hey… I am an entrepreneur, and there is nothing like Sunday for an entrepreneur. I logged in at the same 5:30 in the morning and Boom!

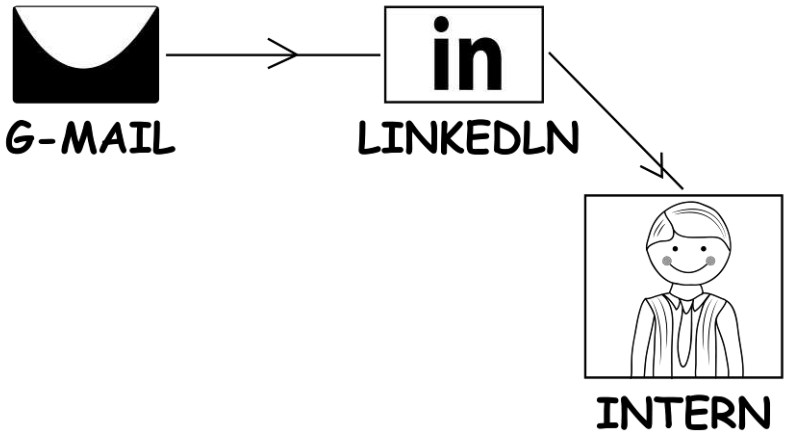
We had access :)

But you see, Only stage 1 was completed. It seems that our Stage 2 and Stage 3 failed for some reasons which I am not sure of. Maybe it was some AV or DLP.



But don't be surprised, My experience in these things is so wide enough that I was already expecting this. This e-mail which I have access to is a personal Gmail address.

This is where my Plan B starts.



I will be resetting her Linkedin Account password using this e-mail which I have access to and then use that account to send a message to the second admin to ask for the latest copy of rules.

To minimize the risk, My plan was to trade off my time. Time is the most crucial asset. We have already wasted more than 10 days now, and I am now going to lose 5 more days.

Why?

If I get access to those Cloudflare rules on Friday night, My team will have 2 days of the weekend for

a minimum to get things done since no one is in office. Secondly, this lady has to go out for a friend's wedding on Friday morning. This was a perfect opportunity.

This is the chat with that intern admin at 5:47 PM in the evening (they leave at 6:00 PM from the office)

> Hey, I got an alert from Cloudflare. All okay?

>> oh thank you so much for checking this. I am not sure what's going on. It's flooded with IP's and all are getting blocked

> I'll be back home in a few hours, send me a copy of current configs. I'll guide you on this

>> On your e-mail?

> No, send me here itself, I don't have my laptop

Now here the trick is that this admin is an intern. As his senior, I already have the element of familiarity & authority. Since the messages are coming from a LinkedIn account, and people treat LinkedIn as a Corporate Platform, We have the element of trust. Finally, With that DDoS attack, I created the element of fear & urgency.

As planned, he fell for it. He sent me pictures of his dashboard, which had a lot of data for my team.
I didn't want to lose this admin here, so we stopped the DDos and I messaged him back that I took care of it.

He said thank you, and I deleted the whole conversation to clear the tracks. By now, I have access to three things.

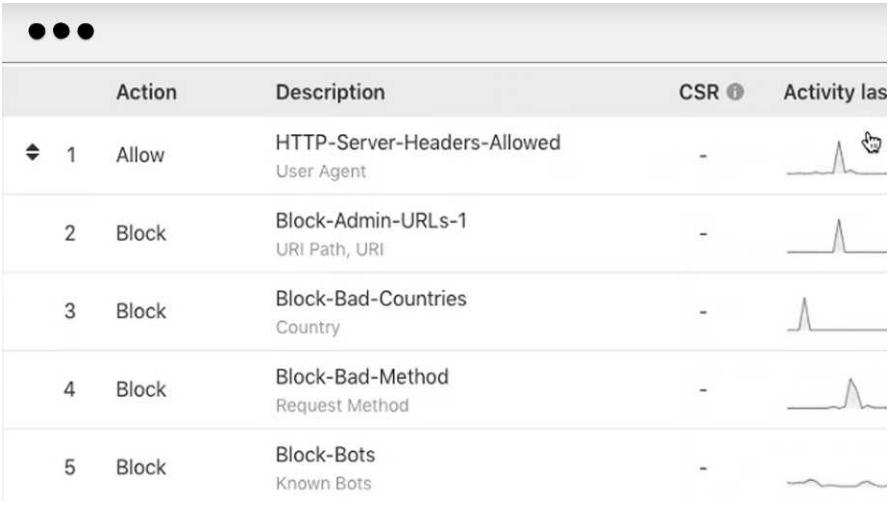
1. Email & LinkedIn Account
2. Dumb Admin
3. Cloudflare Dashboard Pictures

Time to continue analyzing these rules now and see if we have some loopholes.

SECRET #4

MIS-CONFIGURATIONS ARE ALWAYS THERE

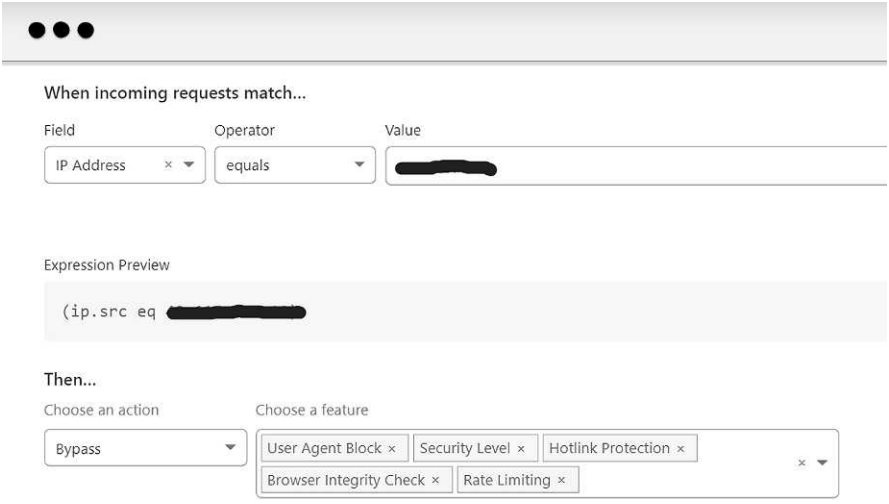
The time when I was thinking that we nailed it by getting those Cloudflare rules



	Action	Description	CSR	Activity las
1	Allow	HTTP-Server-Headers-Allowed User Agent	-	
2	Block	Block-Admin-URLs-1 URI Path, URI	-	
3	Block	Block-Bad-Countries Country	-	
4	Block	Block-Bad-Method Request Method	-	
5	Block	Block-Bots Known Bots	-	

But once we started analyzing these, things went really stretched up.

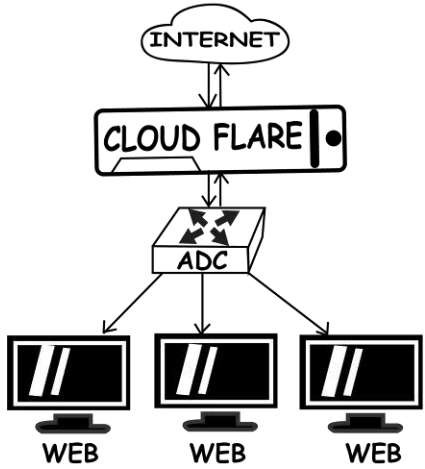
We found out that there was only 1 rule in the whole account which has BYPASS action set. The rule requires a particular whitelisted IP address which these administrators and developers must be getting using a corporate VPN connection.



If we don't have this IP address, what else? Luckily our dumb admin has sent everything including DNS Entries Exposing their Webserver's Real IP Address.

I did the same as you would. Tried to access the WebServer directly using its real IP. As expected, It didn't work.

It appears that there is an ADC in between us and the webserver which will only listen to CloudFlare's IP address. ADC stands for Application Delivery Controller. It usually acts as a Load Balancer, SSL Offloaded, and performs Firewall functions also.



There are minimal brands in the ADC space, Edge Nexus is what I personally like.
(No endorsements by the way)

We are stuck again…
Not a new thing, right?

I asked my team to come on a quick Zoom Call
(Yeah, we use the zoom!)

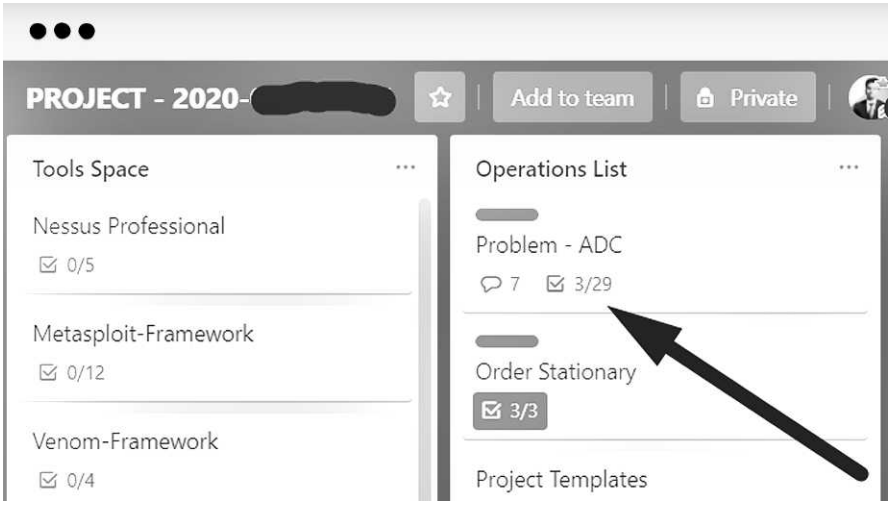
And this is what I said…

It appears that the Web Server only has an internal IP which is accessible to that ADC which only listens and replies to CloudFlare and finally Cloudflare has rules and to bypass those rules, We need an IP address which only comes from a VPN connection and to get that, we have no idea about how to.

Do some brainstorming. See what we have till now. How can we get access?

Think fast But think practical

Everyone is busy with ideas. We are managing everything on Trello.com board which we have created specifically for this project. Everyone is putting new things in comments.



Till evening, we don't really have a decent plan on how to get this IP address.

I still remember that night…
I couldn't sleep because it's the last week of this month and we are not even close to hacking anything.

Next morning, yes you got it right… the same 5:30 AM
I see two members of my team online.

It seems they haven't slept for the whole night! How do I know this… Well, come on… These people never show up before 11 AM. This is 4:30 hours early.

As soon as I was about to say what the hell are you people doing here… One of them says… We are so close!

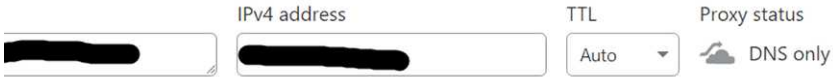
I said what? And then she shows me the progress. This was the most significant moment of this project. That happiness which you can not explain. When you are depressed, and suddenly you get hope.

I know you are dying to know about what really happened. Let me tell you then…

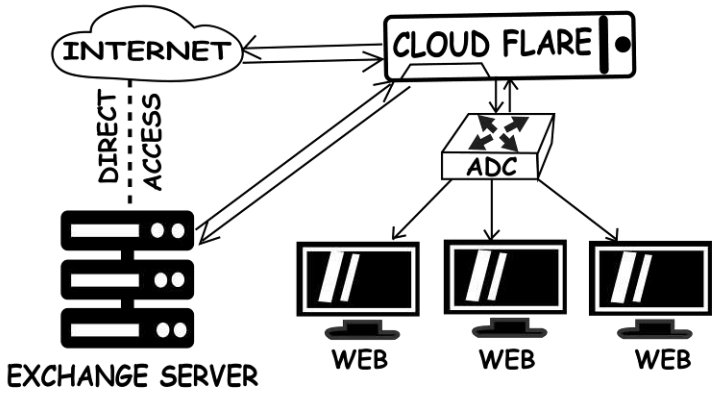
So both of these were working from the whole night as you already know. They were analyzing the data on what we have. They have found an IP.

It appears to be an Exchange Server in the DMZ area. The first rule of Full Stack Security is that you never keep your Exchange & Database Servers in DMZ.

Moreover, It was not locked to listen only to CloudFlare as that record is not proxied.



So by now, our master board looks like this.



The bad part was that again it's real life and nothing is like those silly courses you learn from.

This was a fully updated Exchange 2019 version. There is just no way to hack it. I don't know if it's just me or you must have also felt this too, sometimes when you really want to do something… God snaps his fingers, and a miracle happens.

A 0Day Exploit got publicized by ZDI.

https://www.thezdi.com/blog/2020/2/24/cve-2020-0688-remote-code-execution-on-microsoft-exchange-server-through-fixed-cryptographic-keys

It was a Remote Code Execution. Since it was a 0Day, no IPS of theirs could have stopped this as they didn't have it in their signature database.

Within a few hours of this, hundreds of discord and telegram chats opened and various exploit writers made the code available with their own twists.

Search on google for zcgonvh if you want to see some great stuff. (It should be noted that my team members and I are in no way associated with this word)

Since we don't know about any adverse effects of the code we picked, We decided to test it in our environment first. The analysis resulted in it being entirely reliable and safe to the system.

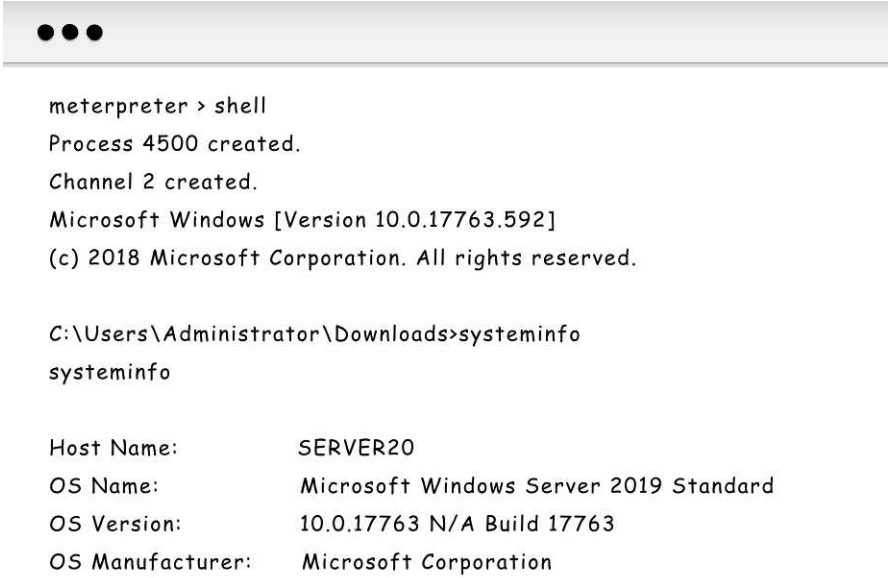
My exploit writer quickly ported it to the Metasploit Framework, which was connected to Faraday Suite, where we were managing the whole project.

Now we just need a valid username and password for this to work. Good news is that you know our Social Engineering skills and if we can fool the security administrator to send us all this access, this is just a standard user account that we need.

It was a simple 1-hour game for us to get the user and password, thanks to our dumb admin again.

Now we had everything. We executed the attack, got a reverse shell on the machine.

It was interesting to see that this Exchange was running on Windows Server 2019 Standard Operating System.



```
meterpreter > shell
Process 4500 created.
Channel 2 created.
Microsoft Windows [Version 10.0.17763.592]
(c) 2018 Microsoft Corporation. All rights reserved.

C:\Users\Administrator\Downloads>systeminfo
systeminfo

Host Name:            SERVER20
OS Name:              Microsoft Windows Server 2019 Standard
OS Version:           10.0.17763 N/A Build 17763
OS Manufacturer:      Microsoft Corporation
```

We quickly launched persistence measures on the system, and now I can finally say...

We did it!

In less than a month, We are in the company with system-level access. We still have 2 months left in our test, and all is good. I need to report this to the client and need to arrange a followup meeting now.

But first… It's Party Time!!!

SECRET #5

IT'S A BUSINESS, CLOSE THE CLIENT OR DIE!

This is something no penetration testing class (apart from mine) teaches you how to handle a client and what to do when it starts going out of your hands.

As you already know, We have system-level access on their exchange server… Did I tell you about that meeting with the client's panel, Yes I Did, right?

On Feb 29th, 2020 (it was a leap year)
I got this e-mail from the CIO who was directly acting as our point of contact for this project.

Dear Mr Bansal,

First of all, I would like to congratulate your team for sending me some robust reports. I am very impressed with your social engineering trick.

It seems that our employees need more realistic training on these advanced vectors. Let's talk about this next week on a video call.

Secondly, I am concerned about your method of getting access. Since a 0day attack was

not a part of our initial rules of engagement, I am afraid that this is raising objections.

Protecting against a 0Day is impossible for any organization. I would request you to kindly consider this out of your scope and find some other way to gain access. I have consulted our whole panel, and this is our final decision.

I am sure you can figure this out.

Name_Here

This e-mail shattered the motivation of my teammates. Just imagine that you are working with me on this project. We have pushed our limits and suddenly all this. How can they just reject our exploit???

On second thought, they are right that this was pure luck. Having a 0Day against a Microsoft Exchange Server is just way off. We could not have created this ourselves, right?

If you think you could, then you have a completely wrong mindset. Maybe because what you have been hearing from people who work on a 9to5 job.

This is a business of big numbers. If I have the capability of creating a 0Day which can remotely exploit any Exchange Server in this world, and gives you system-level access… I can sell that exploit to

someone like Kevin on his Absolute Exploit Exchange for at least a Million Dollars.

Yeah, A Million Dollars…

It's an entirely different market, not for some 9to5 job people. You can not even buy anything until you come from a reference who is a reputed client of theirs. Forget about selling… It's exclusive.

But if I had that exploit, I would sell it to them.
So it was not really practical for a person of that sophistication (Possibly State-Sponsored Hacker Who Made This 0Day) to be working on some project of around $250K. (this was our initial budget)
Then I realized it!
What the hell am I doing?

I am convincing myself that we did get lucky and we used an out of the scope thing… What's wrong with me?

Is it business, right?
I need to convince them and show them that this can be practical.

So I decided to take the CIO on call.
(Never talk on e-mail when you have to convince someone. The phone is the only place where you get the ability to drive the conversation, change their mindset, put pressure, give reasons, listen to them and then turn their objections into actionable decisions that you want them to take)

All my calls are recorded so we transcripted it here for the book. But I have changed the person's name to a fictional character named James. Also, I have only included the essential parts of our conversation here, which you should know and which I am allowed to share as per NDA.

James: Yeah, Sagar, but as I said, it's just out of scope. I don't think I can do anything about this.

Me: Okay, Let me ask you a question… What is the goal of this penetration test?

James: look I know where you are going, but I am telling you, It's not practical. You can not just use an unplanned exploit like this.

Me: James… listen to me… forget this thing for a moment. Let's talk for 5 minutes. After that, if you still think this call was not worth it. I will personally train your employees on that social engineering thing which you wanted to discuss. (this brings skin in the game, Now he will pay attention and also cooperate in answering)

James: You mean you will personally train them for free? Is that what you just said? (Oh yeah, but he is never gonna win me)

Me: Yes! You have my word. Now let's come to my question again… What is the goal of this penetration test?

James: to test our security.

Me: So tell me one thing, Why are we wasting all this time in a Double-Blind Test? Why not change this engagement to blind test? Don't you think that would be much easier for us to work in where your team can alert us when we go out of scope? That would save a lot of time, and since we are lagging behind, We would be able to provide a better quality audit to you?

James: I don't think that's a good idea. We also want to test the skills of our employees. If I wanted a blind test, We could just have used any other firm. Why would we hire you, then? (Here is the truth)

Me: Hmm… that's correct. So basically what you are saying is that you want to test your security measures but also your team capabilities at the same time to defend the organization right… Am I Correct?

James: yes

Me: I am sorry what? (I heard him already, but you should ask these questions whenever they say something that you really want them to accept and reinforce)

James: Yeah, I said, that's what we want to do.

Me: Right… So James, don't you think that your team should have defended against this attack of ours. Why were they not looking at their SEIM logs? Why was that Exchange Server not configured

correctly to only accept requests from your ADC as you do with every machine? Why do you think your Exchange Server even allowed a connection out to our computer. There was no DLP on that system to block the connection.

James: Hmm… (deep thought) but still, if there was not 0Day, then you couldn't have hacked it. We are always on the same page as Sagar.

Me: yes, I agree with your point, James. I am not saying that this was a balanced attack. But let's assume for a moment, If we would have done this same thing with a known vulnerability, would this be a critical issue for you?

James: yes, in that case, I suppose

Me: Sorry, you are breaking. **(**he wasn't but you the trick right?)

James: oh! Can you hear me?

Me: yes, I can.

James: So, I was saying that if this was a known vulnerability, then yes, this would be a critical issue for our security.

Me: Right... So now James, let's do one thing, You don't tell this to your team. Give them like a week. Let's see if they figure it out themself. If they do find it and they patch it. I will consider this out of scope, and we will find some other way.

If they don't find it, then we will not consider this as out of scope. Of course, they don't have an official patch from Microsoft, but they need to fix the issue using some temporary fix right? A week is a long time for a publicized exploit to be patched against.

James: Hmm, but what will you do till then?

Me: yeah, let us even keep digging deeper. See if your team can find us. This could be an excellent exercise for them. Who knows if they patch this and we consider this out of scope, and we never get into your security defences. We won't be able to test their Detection Skills. This way, you will get a double benefit. What do you say?

James: Yeah, that makes sense. I will talk to the board and come back to you.

After that, we had some more talk, and I sold him the social engineering training program for $2000 per person (it's a 10 days live program, so I took 30 key employees of him totalling a $60K + all my expenses of travelling, stay, etc. etc.)

Now what?

I just need to waste 7 days. Why Waste?
Yes. Because if we do something, even one of their admin sees something on a log, they will know the vulnerability and patch it which means we lose the terms of this new engagement and this 0Day goes out of scope. So it is better to simply waste 7 days

and even clear all the scripts and backdoors we installed & removed all the logs we could.

An important note which a lot of 9to5 red teamers won't know is that this is not at all-sufficient. If you think cleaning these logs from the machines hacked will fix the problem, then let me tell you this,

In an Enterprise-Grade Security, every security device is connected to a Central Logging System which can be a SEIM or a more advanced SOAR solution. If they are doing their security properly (Which I Know They Are), Some admin will definitely see us.

We need to keep them busy so that they ignore this Exchange Server. This is what we did…
We started various fake attacks on their other main website. XSS, SQLi, Directory Transversals, Ddos, and everything we know will not work.

The purpose was never to get it working, It was just to fill their monitors with so much data that they can not handle, and these logs of our attack get buried in all this noise.

We deliberately attacked using 100 different machines, How did we get all these? Simple… We leased them on Amazon Web Services for a week. Setup one machine to change IP every minute and perform automated attacks and simply cloned that VM Instance 99 more times.

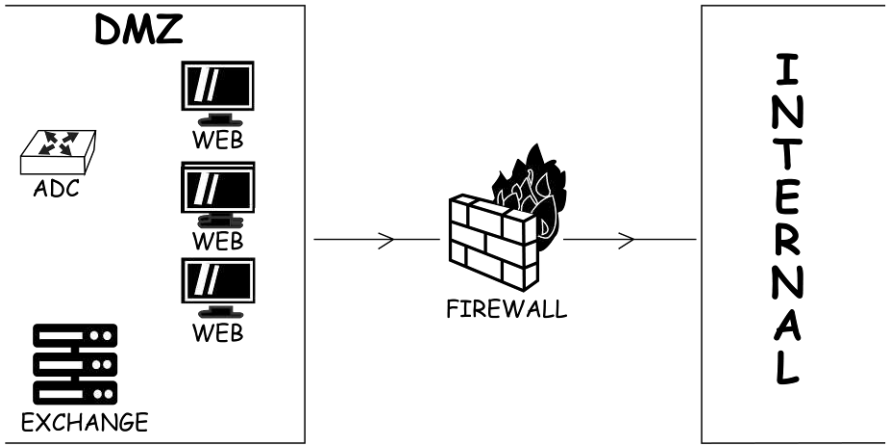
Fast forward to a week, Another call with CIO and Boom!

A Green Signal. That's what I wanted.
Shall we go to a movie now?

SECRET #6

DON'T FIND HOW TO DO IT, BUT WHO CAN

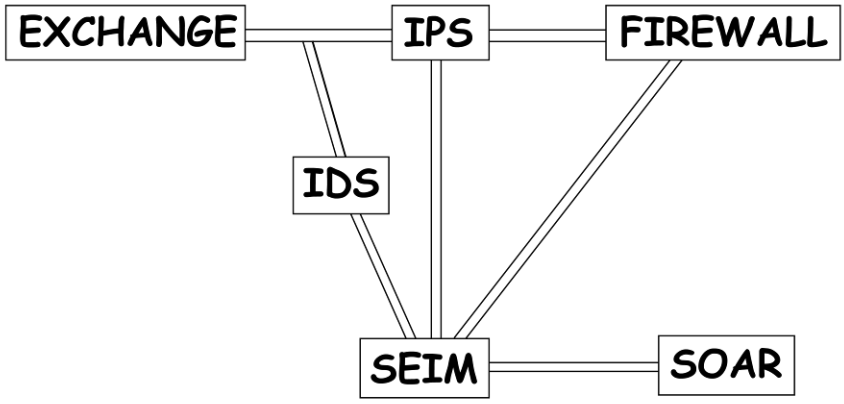
Now you may be thinking that everything is fine now. We got our green signal, we are in the DMZ. Let's start attacking the internal network.



Hold on, cowboy!

The fact is that we are in a Demilitarized Zone. There is a reason why we also call it a Jail Box :)

You see, we are basically locked in this network area. There is a firewall which will separate us from the internal network. Think of it as your Jail bars.



There is an IDS which is sniffing all the traffic passively and raising alerts in that SEIM system of theirs. An IPS or SOAR which will be blocking all our exploit attempts. Think of these as Police & Guards.

Did you get it? A Jail Box... Huh?

So what do we do?

Of course, we need to get through that Firewall by finding some loophole in its configurations.
But after wasting a few hours, we discovered that this machine didn't have any policies configured that would allow inbound traffic through DMZ to Internal.

All of this makes a lot of sense now!
It seems like this server is not in production…

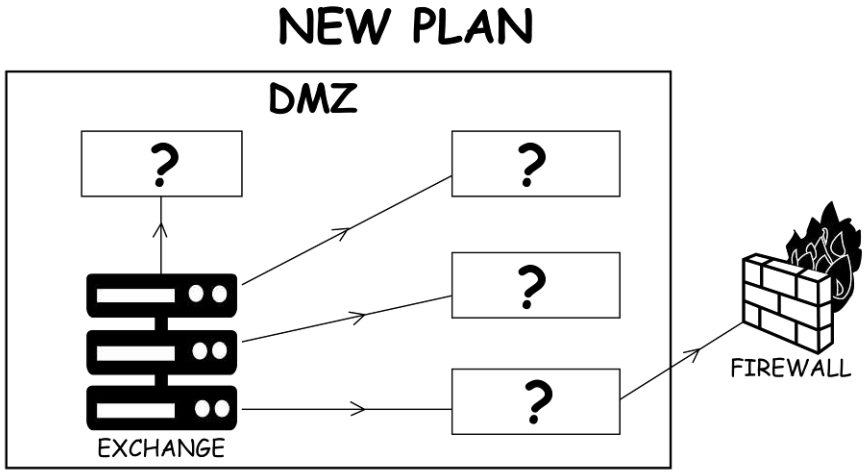
Security Admins are still configuring it.

You may ask what now?

Well, we are locked in one network segment.

So let's see what other machines we have, compromise one of them, and probably we may find some policy in the Firewall, and hopefully, some traffic will be passing through it.

That's the only shot we can take at the moment…



So we started a very slow host discovery scan. Yeah… slow like it took a whole day to map that one segment. Including the night!

We know there are 6 more servers apart from us in this network. It took a whole day just to check this. We need to speed up but without creating any noise. Hence this is what I sent to my team.

Guys,
Check only standard ports we think should be open. Think what a DMZ can host?

Web Server - Port 80 and 443.
Database Server - Port 3306 and 5432
Email Server - Port 25, 110, 456, 587
FTP Server - 20, 21
SSH Access - 22

These are only 11 ports. If we are scanning 6 machines, we only need 66 requests. Each request delay is 5 minutes. So we need 330 minutes which is only 5 and a half hours

,
You may be thinking ah… another day is gone. Don't worry. This will pay off. I have faith in my team!

So by evening, we have all the systems mapped as per these ports. Three of these are Web Servers. One of them was an FTP Server. One was a Mail Server with all e-mail Ports open as specified. The last one was undetected, I have a strong feeling it must be a Database server because how can they not have one?

But then I thought that they may have a Database server in the internal network; instead, it can be a Honeypot machine.

Since we used only selected ports so we can not be entirely sure of this last machine without doing a full port scan. (and the fact of it being a honeypot makes me shiver. I am not gonna check this at all)

Now, as the last machine is out of the target database, we need services and version numbers on the remaining machines. This is easy. Since we know which machine has which port open, we only need 3 requests for port 80 on web servers, 1 for port 21 for FTP server, 1 for port 25 on the E-mail server. In total, 5 Requests.

Another 25 minutes gone and now we have the versions of these services as well. Time to check if any of these is vulnerable or not. It's doubtful in such a high-end environment but let's give it a try tomorrow.

After having a sweet sleep, Again 5:30 sharp, I started looking at version numbers on various databases. Unexpectedly I found a vulnerability in the machine I was least expecting on.

It was that Mail Server we found. Maybe they have been using it, and now they are slowly moving to the Exchange Setup which we already hacked. Anyways, it was running Exim 4.92.2, and when I searched this version of Exim on CVEDetails.com, I found that it has an awful history of exploits.

This specific version of ours is vulnerable, and 3 Different CVE's have been issued in 2019. Two of them are with a score of 10. Giving complete access to the system using RCE.

Well, what's there to think then. Let's fire an exploit.

We fired it, and nothing happened xD
As expected, some HIPS, DLP or AV must have blocked it!

Now a lot of newbies will come and say, but there are techniques to bypass these protections.
You see, the challenge is not in bypassing. It is always the Detection Mechanisms. Don't you think if our attack has been blocked, it must be on their SEIM by now?

I don't know anything from here. It's not my area of study. I have no experience in bypassing this stuff. But luckily, We don't do pentesting as a one-man army job. It's a team project, and my team has an expert who specializes in bypassing these protections.

He was trying everything and filling our Proprietary Methodology Template. This template is based on SANS Whitepapers. I was getting these report sheets as he was filling them while doing his tests.

The first one I got was Obfuscation. Then Simple Fragmentation, Next was Decoy Trees,

Nothing was working.

I was feeling so trapped in this. My throat was getting dry, eyes were numb. Stomach was getting an anxious feeling like acid was filling it in. I was fainting and wanted to run away from this.

He was just trying all possible tests. He tried Encoding, Wrapping Sequence Numbers, Decoy Messages, and finally, I see a command shell!

```
[*] Started reverse TCP handler on 0.0.0.0:1222
[*] Command shell session 17 opened (IP_Removed_Due_To_NDA)


systeminfo
/bin/bash: line 7: systeminfo: command not found
uname -a
Linux fedora-s-lon1-01 5.0.16-300.fc30.x86_64 #1 SMP Tue May 14
19:33:09 UTC 2019 x86_64 x86_64 x86_64 GNU/Linux
```

I lifted up the water bottle in front of my desk and drank it completely. All empty in one shot. My stomach started getting normal, and I was now feeling much better.

I checked the latest test he did. It was a custom setup of NDR Flags with Big Indian Encoding of our exploit. That's what the paper in his hand says at least!

I seriously don't understand what that means, how he did it, and why it got successful. It's just not my work. But I am happy to know that now we control this mail server.

Did we alert them? Hmmm, I don't know…
I guess all of these blocked attacks must have been reflected somewhere. But at the moment, I have no way to find that out.

But it was not a meterpreter shell, and we really need one. We tried a quick post-exploitation module, but it failed.

After all this fantastic work, I don't think anyone in my team had any energy left. So maybe we can try it later.

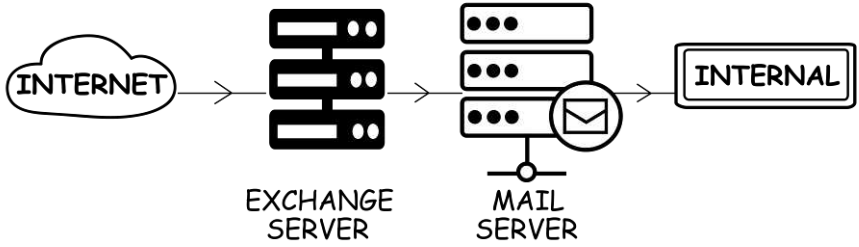
They need rest.
Let's call it a day.

SECRET #7

THE LEVERAGED SHELL METHOD

I believe it is essential to revise the current situation once. We have hacked the Exchange Server through which we have pivoted and hacked Mail Server and now using this we are trying to pivot to the internal network.

Oh, and the latest update is that we have meterpreter running on the mail server thanks to my teammates.



After spending some time on Mail Server, it seems like there are many ports we can use for our attack. Port 25, 110, 456, 587 for E-mail Services, Port 53 for DNS and finally Port 389 for LDAP are marked whitelisted in the Firewall towards our internal network. Maybe they had more, but at the moment, we could only see this much.

Now it's the time to use what I call The Leveraged Shell Method (Yeah, it's not an industry term, I created it myself)

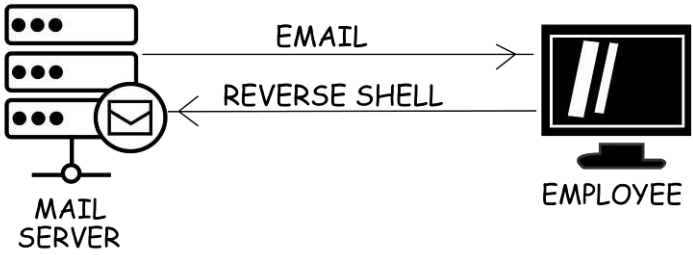
It is interesting to say that I have learnt this technique from OFFSEC. They have a report on the internet which is about hacking a fake company. Their method was impressive but there is a harsh reality behind it.

I think they made it highly unrealistic where they get direct access to the internal network without routing the attack from DMZ. This is something which I learnt in real testing over the years

Things in a Real Enterprise Environment were complicated, with a lot of security devices. I'll show you the actual situation.

Basically, the plan is to use this Mail Server as a Command & Control Center and send a reverse shell payload to some employees using e-mail.

Since I own this e-mail server, it was super easy to send an e-mail from their domain to employees.



The challenge is going to be how to bypass its internal network endpoint security system.

I asked my exploit writer to create a meterpreter reverse shell payload which is embedded in a pdf file. (The same old technique) and use the same methods which we used to bypass protections for this e-mail server in the last chapter.
This Payload should send us a reverse connection on port 456 since it is opened in the Firewall.

Why 456 and not any other port like 25 or 110?
We just saw the port usage logs, and it shows port 456 is rarely used. So it seems to me that we can disable the service which is listening on this port and instead use it to port forward the reverse shell back to our exchange server which can then port forward that reverse shell back to our machine.

It's that simple…

So we sent an e-mail using this mail server to a random user of the company. (thanks to LinkedIn again for the e-mail)

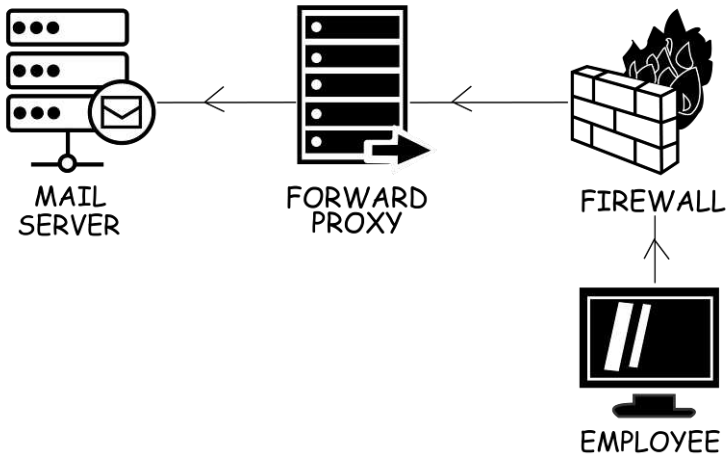
Nothing happened.
Maybe he/she didn't click on it?

We sent another e-mail to another user and left it for a day waiting for him to click on our Payload…
Nothing happened again!

Is our Payload not working? What's the problem???
They say, the third time the charm, so let's do it once more…

Still, nothing happened!

After thinking for some time, I remembered that device in DMZ which we left thinking as a Honeypot. I think it was not a honeypot but rather a forward proxy.



Most enterprises use a dual forward Proxy setup where they have an internal forward proxy which resides in the DMZ area and an external forward proxy which is placed outside the Firewall.

The setup allows secure communication because users are never connecting to any device directly as they are going through the proxy.

That explains why we are not able to get any reverse shell on a system because these users may be set to use a specific port on their proxy to communicate with the e-mail server and we have no idea about that, so it seems we are kind of locked again.

The next day we got the idea that maybe the admin department would have Direct Access to DMZ. They need it to manage all of these devices daily, and they cannot go through a proxy every time, right?

So we decided to take a chance on a GRC Admin. We sent the same e-mail with the Payload.

Nothing happened here as well.
This was a point of time again when I was kind of crying from inside and everyone in my team too.

But my experience is the best asset I have developed over the years. We sent this e-mail to that admin.

From: r.shane@target.gov
To: ouradmin@target.gov
Subject: Urgent help needed

Body: I am not able to open this file. Please help me asap, We have a meeting starting in 1 hour!!!

You may say what's so special about this e-mail? Looks pretty simple, huh?

Here is the twist, We didn't send any file in the attachment!
And as expected, he replied by saying that it seems like I forgot to send the file (just for a note, it was not even his department to help me but you know how helpful people can be right?)

As soon as he replied, I checked the e-mail reply. It's signature, the font used, the headers of the e-mail and I realized that it was Ubuntu Linux.

This was the whole problem. Our assumption was that operating systems will be Windows and our Payload was made accordingly, but where the operating system is Linux.

Once we knew this, It was a five minutes work to remould the Payload for Ubuntu and send it again.

```
[*] Started reverse TCP handler on 0.0.0.0:456
[*] Command shell session 4 opened (IP_REMOVED)


uid
id
uid=0(root) gid=0(root) groups=0(root)
uname -a
Linux ubuntu-s-1gb-lon151-15 5.4.0-29-generic #33-Ubuntu SMP Wed
```

And it worked.
We have access to his computer now.

It was the time to set up some persistence scripts, and before we leave the system here, I decided to deploy a keylogger.

The challenge was that it is Ubuntu Machine and Metasploit really doesn't have keyscan_start for this. So the only choice was to install a keylogger directly on the system.

```
meterpreter > shell

Process 12211 created.

Channel 5 created.

sudo DEBIAN_FRONTEND=noninteractive apt-get install -qq logkeys < /dev/null > /dev/null

touch /var/log/accessk.log

sudo logkeys --start --output /var/log/accessk.log

tail -f /var/log/accessk.log

Logging started ...
```

This month is almost over, and I need to prepare for our monthly report.

In the meantime, My team can keep an eye on all the inputs that come from the keyboard of this admin.

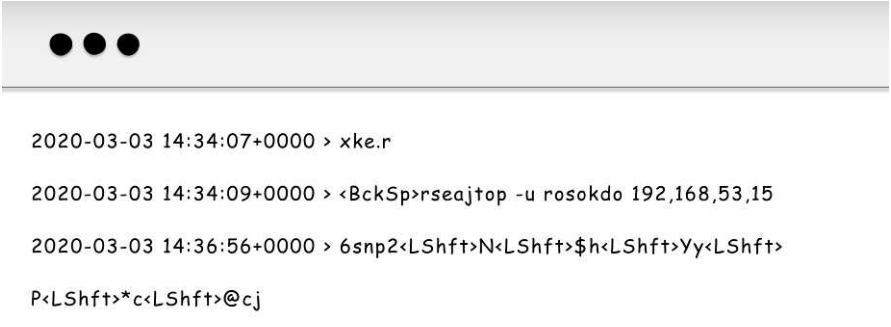
Time to face the board!

SECRET #8

OBSERVATION IS A GREAT SKILL YOU NEED

Last time, We managed to get a reverse shell back from an admin computer which was running Ubuntu. Before leaving the system, I installed LogKeys Keylogger on it and kept a tail session running so that we can see whatever is being typed.

Soon after, this admin did an RDP Login to another machine using his credentials which we got using our Keylogger.



Huh… Not too clean right?
It seems they were using a different keyboard layout or something was messing up these logs.

But if you analyze them carefully, you can see that the first command is clear which is being logged as xke.r

So we made our mapping table…

- c is being replaced by x
- l is being replaced by k
- a is being replaced by dot (.)

Note that e and r do not get affected.

Similarly, the next command looks like rdesktop -u username syntax.

Analyzing the first-word rdesktop which was logged as rseajtop

- **d is replaced by s**
- **s is replaced by a**
- **j is replaced by k**
- **t, o and p are not replaced**

Following the sequence, -u is logged as -u which means

- **- and u are not replaced**

Then there is username rosokdo.
Using the rules we had till now, We can predict…

- **r never gets replaced**
- **o never gets replaced**
- **s means a**
- **o never gets replaced**
- **k means l**
- **d we don't Know**
- **o never gets replaced**

So that should be roaol?o
That really doesn't make sense… isn't it?

After analyzing it for some time, we found a pattern….
Not completely concrete but quite satisfying. Each letter is replaced by the letter before it, as per American QWERTY Keyboard.

- **c - x (letter before)**
- **l - k (letter before)**
- **a - . (no relation)**
- **d - s (letter before)**
- **s - a (letter before)**
- **j - k (letter after)**

We had only 2 exceptions which were a and j where a didn't have any meaning and j was getting replaced by letter after it instead of letter before it.

We found one more amazing observation that first line of the keyboard which has letters QWERTYUIOP is not being replaced because as you can see above,

letters e,r,t,u,o,p were not replaced, and all are first-line characters on the keyboard.

Anyways, I used a tool called Cryptool to create all possible combinations following these rules.

The one which caught my eye was rodolfo which was made as

- **r remains r (first line letter)**
- **o remains o (first line letter)**
- **d gets replaced by s (letter before)**
- **o remains o (first line letter)**
- **l gets replaced by k (letter before)**
- **f gets replaced by d (letter before)**
- **o remains o (first line letter)**

Hence rosokdo means rodolfo which is an American/Spanish name meaning Famous Wolf

I found this meaning
https://www.sheknows.com/baby-names/name/rodolfo/

You may say what's the need to check it's meaning… But believe me, When something takes so much time and energy, it kind of becomes mandatory!

Anyways, coming back to the track, This is the first time I am disclosing an IP address with a SPECIAL PERMISSION from my client. 192,168,53,15 is extremely simple that dot (.) is being replaced by comma (,) so it means 192.168.53.15

Similarly, we kept going, and the next line which should probably be the password looked really hard.

6snp2<LShft>N<LShft>$h<LShft>Yy<LShft>P<LShf
t>*c<LShft>@cj

You can see this line has <LShft> which is probably being used to Capitalize the letters. Hence making its rules was quite hard, but we generated over 82 passwords following our patterns.

82 is quite a small number to brute force, but we wanted to go slow, and it was already evening time. If we RDP at this time, it may trigger the systems as every failed attempt will be marked with a Windows Error and will reflect in SEIM, So we decided to leave it there for the day to continue tomorrow.

The next day we decided on a one password per ten minutes brute force. This should try 6 passwords per hour which means it will take around 13.6 hours max, and if we are lucky enough, it may just take 30 minutes working the first 3 passwords.

Lol, I don't know how I even say these things so easily… It took more than 7 hours to get the password and these 7 hours were horribly long.

Every time it tried a password, the fear of someone noticing and then the fear of automatic account lockout was killing us.

The password was 6dmp2M$jYyP*v@vK

If you compare

6snp2<LShft>N<LShft>$h<LShft>Yy<LShft>P<LShf
t>*c<LShft>@cj

Which means 6snp2N$hYyP*c@cj (Ignore that <LShft> which is already being reflected in the letters)

Came out to be 6dmp2M$jYyP*v@vK after decoding.

Shall we call it a day?

SECRET #9

STAY PASSIVE, AND DATA WILL COME

You may be thinking that now we will go ahead and attack DC. That is where the majority of noobs will go wrong.

Penetration Testing is an objective-oriented process. Our objective, as decided in advance with the management, was to gain access to proprietary information.

Yes, compromising a DC is a potential option, maybe if we had a much larger time frame, we could have gained access to the DC. But at the moment, we had less than a month, and multiple detection systems since this is an internal network.

So we decided to use the Lateral Movement as our best shot to keep increasing our access range.

We kept sniffing passwords, keylogging, taking screenshots whenever possible, sometimes even recording audio.

It was a game of passive listening.
The tool we utilized here was Empire and Metasploit in combination.

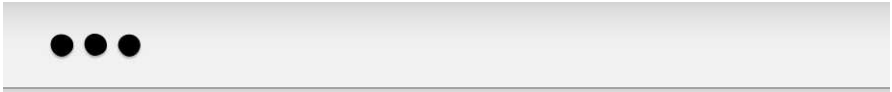
Empire always impresses me with its capabilities of acquiring a Stable Session for Lateral Movement and all the modules it comes with.

```
(Empire: powershell/lateral_movement/invoke_smbexec) > set Listener http
(Empire: powershell/lateral_movement/invoke_smbexec) > set Domain (REMOVED)
(Empire: powershell/lateral_movement/invoke_smbexec) > set ComputerName adm-grc-fw-02
(Empire: powershell/lateral_movement/invoke_smbexec) > set Username (REMOVED)
(Empire: powershell/lateral_movement/invoke_smbexec) > set Hash (REMOVED)
(Empire: powershell/lateral_movement/invoke_smbexec) > execute

[*] Tasked 51PSSD6N to run TASK_CMD_WAIT
[*] Agent 51PSSD6N tasked with task ID 2
[*] Tasked agent 51PSSD6N to run module power-shell/lateral_ movement/invoke_smbexec
[*] Sending POWERSHELL stager (stage 1) to (IP_REMOVED)
[*] Agent 51PSSD6N returned results.
Command executed with service REMOVED  on adm-grc-fw-02

[*] Valid results returned by (IP_REMOVED)
[*] New agent 6L2FE1BA checked in
[*] Initial agent 6L2FE1BA  from (IP_REMOVED) now active (Slack)
[*] Sending agent (stage 2) to 6L2FE1BA  at (IP_REMOVED)
```

Sweet, After this, it was quite easy to run mimikatz and other modules and get users and hashes.

NDA is stringent for this specific chapter; hence I can only show you these half-visible images.

Here are all the machines we compromised to level up towards DC administrator access.



```
(Empire: powershell/lateral_movemont/invoke_smboxec) > agents

[*] Active agents:

Name        La  Internal IP   Machine Name   Username    Process     PID
51PSSD6N    ps  REMOVED       adm-grc-fw-02  REMOVED     PowerShell  513
6L2FE1BA    ps  REMOVED       adm-grc-fw-02  REMOVED     PowerShell  3411
NCTDLQ7P    ps  REMOVED       REMOVED        REMOVED     PowerShell  3660
BQ3FE54R    ps  REMOVED       REMOVED        REMOVED     PowerShell  635
S6T755PC    ps  REMOVED       REMOVED        REMOVED     PowerShell  2492
8RRA7C34    ps  REMOVED       REMOVED        REMOVED     PowerShell  814
9TWV4CDJ    ps  REMOVED       REMOVED        REMOVED     PowerShell  416
```

This is how our Hashes looked

```
[*] Valid results returned by (IP_REMOVED)

(Empire: powershell/credentials/mimikatz/logonpasswords) > creds


Credentials:

CredID  CredType  Domain     UserName       Host       Password
1       hash      REMOVED    adm-grc-fw-02  REMOVED    REMOVED
2       hash      REMOVED    REMOVED        REMOVED    REMOVED
3       hash      REMOVED    REMOVED        REMOVED    REMOVED
4       hash      REMOVED    REMOVED        REMOVED    REMOVED
5       hash      REMOVED    REMOVED        REMOVED    REMOVED
6       hash      REMOVED    REMOVED        REMOVED    REMOVED
7       hash      REMOVED    REMOVED        REMOVED    REMOVED
```

Anyways, What I am trying to teach you is that Owning DC is not the only thing. Objective matters and our objective was to gain access to as much as proprietary information possible.

We did all of this in a passive mode. Don't think you can just brute-force the DC with some hundred thousand passwords and no one will notice you.

Keep the simple sniffing and keylogging approach. Just listen carefully, and you will get the information yourself.

We collected more than 20 GB of data consisting

1. 153 Screenshots
2. 88 Webcam Captures
3. 14.5 hours of Audio Recordings
4. Thousands of words in Keylogger

I genuinely want to show you all of this, but what can we say. If anything comes out in public, my whole team is going to Jail.

Oh with Jail, I just remembered, You may feel that there are 9 chapters in this book. Why not 10 to make it even. Well, there is, in fact, Chapter 10 which is Unseen and Unedited.

The chapter which was never published. The section due to which we almost got arrested. The Biggest Secret To Nail Any Double-Blind Penetration Test.

I know you want it; in fact, you need it…
Maybe if you check pentest secrets dot com slash ten you may find something interesting.

~ Sagar Bansal