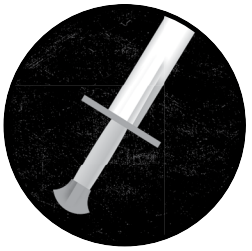
2

BURIED ALIVE



Now that we've properly set up our hacking infrastructure, let's discuss our target for this hacking exercise. We are going after G&S Trust, a niche company that specializes in the offshoring business. They help the wealthiest 1 percent create shell companies in various parts of the world to optimize their asset allocation and revenue streams. What you and I would bluntly call tax evasion is stretched into a whole sentence of obscure financial jargon that makes it sound like an innocent Sunday hobby.

Buckle up people, we are about to embark on a hacking journey that will take us as far as the Seychelles islands, Cyprus, Hong Kong, and other tax havens! Our initial aim is to find out as much about G&S Trust as we

can, as well as any close partners that might give us an in. This reconnaissance phase is all about perseverance: we'll hit some dead ends, but there's always another path to try.

Establishing Contact

Quick, what's the first thing that pops into your head when you think about penetrating a company's defenses? Please don't say Nmap!

Exactly: phishing. If you're looking for the surest way to breach a specific company, as opposed to casting a wide net to catch low-hanging fruit, phishing is a formidable attack strategy. It exploits a basic human weakness, boredom at work, coupled with the infamous see link/click link syndrome.

Phishing may seem like an easy endeavor because of the attention it gets in the media and from self-proclaimed security evangelists, but executing a *clean*, undetected, and targeted phishing campaign requires hard work and meticulous preparation.

Before diving into the technical details, let's first gather a bit of information about G&S Trust. Their main website, *www.gs-crp.com* (Figure 2-1), states that G&S Trust has five senior partners, spread across five geographic locations: Cyprus, the Seychelles, Hong Kong, Malta, and the newly opened Singapore office.

Figure 2-1: The home page for www.gs-crp.com

We might be able to leverage this geographical distribution in our phishing email. For example, staff in the Singapore office might find a new IT tool or problems inherent to new offices convincing subjects.

The main issue we presently face, however, is the small number of potential targets. So far, we've only identified five senior executives, plus a few accountants that popped up on old job postings, which brings our total to a whopping seven. We haven't managed to find a full list of current employees on LinkedIn or similar professional social media websites, so we can't be sure of the total count. This is understandable given the company's line of work and obvious need for secrecy, but it puts us in a minor predicament. Phishing is, after all, a numbers game. The more targets we have, the more flexibility we gain. Seven people is a seriously small pool of targets; it doesn't give us much room for testing, much less failure.

Furthermore, it's not like we can breach the company's network by sending a trapped attachment to senior executives, who likely spend most of their time working on iPads or iPhones from an airport's VIP lounge. Even if we

landed the perfect zero-day on iOS, which arguably would require a significant effort, we'd still be constrained to that specific device, without much room for pivoting to other machines containing the information we're after.

"HACKING IS EASY!"

At some point, you may come across some old-school veteran pentester on social media loudly proclaiming that "Hacking is easy. You just need to drop a USB Rubber Ducky in a parking lot and wait for shells to pour in. Haha."

Right.

Maybe that was true 25 years ago, when you could dial in to an electric main grid over Telnet with "root/root" credentials, but those days are long gone. That's not to say dropping USB keys never works anymore. The WHID project (*https://github.com/whid-injector/WHID/*) is a great testimony to that fact. However, there are some intricate problems to solve for it to be successful: How will you drop these keys anonymously at the target location? What if the target is on the other side of the planet? What if everyone at the target is running USB-C computers? What if the staff are using macOS Catalina, which explicitly asks them to register the keyboard when they plug it in? What about USB whitelisting, enforced in many banks? My point is that one needs to properly plan for all these scenarios and move carefully. Arguing that phishing or USB dropping is an easy and foolproof method is neither sensible nor helpful.

Does that mean hacking is hard? As with so many aspects of life, often the answer is *it depends*. It depends on the company's security posture; their awareness of the implicit risks they accumulate every time they take a shortcut; their willingness to invest time and resources in tech projects with no immediate returns; or, to phrase it differently, their willingness to sacrifice short-term rewards for sustainable and secure growth. Of course, it also depends on the hacker's experience and strategy.

In the case of G&S Trust, we'd love to jump in and give phishing a try, but the simple fact is that the target list is too narrow to guarantee a decent success rate, and managing partners are not the ideal victims for our current scenario.

Let's leave this phishing thing aside for now as a last-resort weapon to try if everything else fails. Maybe we can find another entry point—perhaps a vulnerability in one of their internet-facing applications? Let's dig into that.

Scouring the Web

We use the `dig` command to return the IP address of *www.gs-crp.com* and a `whois` lookup to figure out who hosts the main G&S Trust website (Listing 2-1). The `+short` flag shortens the output of the `dig` command.

```
root@FrontLine:~# dig +short www.gs-crp.com
50.28.34.195
root@FrontLine:~# whois 50.28.34.195
NetRange:       50.28.0.0 - 50.28.127.255
CIDR:           50.28.0.0/17
NetName:        LIQUIDWEB
NetHandle:      NET-50-28-0-0-1
```

Listing 2-1: Inspecting gs-crp.com

The main website is hosted by the third party LiquidWeb, a popular managed web hosting provider. We won't bother checking for vulnerabilities in this website since we're trying to breach G&S Trust's network, not that of its hosting provider.

Instead, we'll look for other websites and applications belonging to G&S Trust. For this we could rely on traditional subdomain enumeration using tools such as DNSRecon (*https://github.com/darkoperator/dnsrecon/*) or Amass (*https://github.com/OWASP/Amass/*), but there's a much more efficient tool to uncover obscure subdomains linked to a company's name: Censys (*https://censys.io/*).

Censys is a platform created by researchers at the University of Michigan that scans the internet looking for open ports, HTTP banners, and other valuable information. In this respect it's no different from Shodan (*https://shodan.io/*), but it gets more interesting than that. Censys also indexes SSL certificates, including the Common Name (CN) field that specifies alternative subdomains covered by the same certificate. This should give us the list of subdomains attached to a current IP address. For example, a simple search for *gs-crp.com* reveals a rich set of subdomains, shown in Figure 2-2.

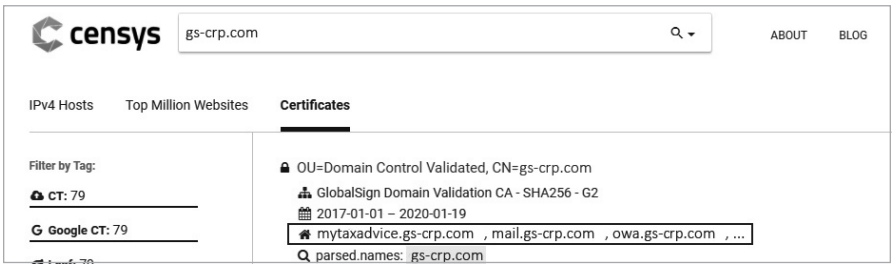


Figure 2-2: Uncovering subdomains with Censys

NOTE *While Censys is a great tool to find the more complicated and uncommon subdomains, such as* mytaxadvice.gs-crp.com*, it's also a good idea to try traditional DNS brute-forcing tools like DNSRecon, to draw the most accurate picture possible.*

The Python script *censys_search.py* available in the book's resources (*https://github.com/sparcflow/HackLikeALegend/*, under the folder *py_scripts*) will query the Censys API, weed out unrelated domains, and clean up the result

so it's in a readable format. It requires free Censys API credentials, which are available for registered accounts from the main website. To obtain a Censys API key:

1. Go to *https://www.censys.io/register/*.
2. Create an account. You'll be sent an email with a confirmation link; click this link to verify your account and log in to the site.
3. Go to *https://search.censys.io/account/api/* and copy the **API ID** and **Secret** values.

Once you have those values, set them as environment variables using the `export` command and run the script on *gs-crp.com*:

```
root@FrontLine:~# export CENSYS_ID=de0ef7de-badd-5...
root@FrontLine:~# export CENSYS_SECRET=eDCQE...
root@FrontLine:~# python censys_search.py gs-crp.com

INFO:__main__:Looking up gs-crp.com on censys
INFO:__main__:Found 32 unique domains
mytaxadvice.gs-crp.com
career.gs-crp.com
mail.gs-crp.com
owa.gs-crp.com
www.gs-crp.com
```

This gives us a tiny list of unique domains associated with the main G&S Trust site. Isn't that depressing? I know G&S Trust is a niche company, but there are local shops with more domains than that!

In case we've missed any associated websites we fire a few more search requests through our custom Censys script, aiming randomly at other potential top-level domains or name variations such as *gs-crp.net*, *gs-crp.org*, *gs-trust.com*, *gs-foundation.com*, and so on, but we get no hits. We end up with the following list:

- *mytaxadvice.gs-crp.com*
- *career.gs-crp.com*
- *mail.gs-crp.com*
- *owa.gs-crp.com*
- *www.gs-crp.com*
- *gstrust-foundation.org*

Some of these websites might be hosted by third parties and others by G&S Trust itself, so just like before, we'll inspect the network owners to shed light on the assets' physical locations.

In the book's resources (*https://github.com/sparcflow/HackLikeALegend/*, folder *py_scripts*) you'll find a handy Python script called *query_whois.py* that exposes the site hosts for a list of domains. It loops through multiple `whois`

calls and extracts relevant information into a readable CSV file. Listing 2-2 shows the results when we run it.

```
root@FrontLine:~# python query_whois.sh domains.txt| column -s"," -t
www.gs-crp.com liquidweb    58.28.0.0-58.28.127.255   US
career.gs-crp.com  liquidweb 58.28.0.0-58.28.127.255  US
mytaxadvice.gs-crp.com liquidweb 58.28.0.0-58.28.127.255 US
gstrust-foundation.org google-cloud 104.19.0.0-104.19.255.255 US
mail.gs-crp.com GS-TRUST 182.239.127.137-182.239.127.145 HK
owa.gs-crp.com GS-TRUST 182.239.127.137-182.239.127.145 HK
```

Listing 2-2: The results of running query_whois.py *on our list of domains*

Only the webmail addresses *mail.gs-crp.com* and *owa.gs-crp.com* seem to be located within the company's IP range that we retrieved in Listing 2-2: 182.239.127.137–182.239.127.145. Other web applications, such as *career.gs-crp.com*, are hosted on cloud providers like LiquidWeb and Google Cloud.

We visit *mail.gs-crp.com* in the browser and notice it is running the Outlook Web App (Figure 2-3)—probably the Microsoft Exchange 2019 version, from the looks of it.

Figure 2-3: We see the Outlook Web App when visiting mail.gs-crp.com.

We try playing with the webmail interface, looking for hidden directories, injecting special characters here and there . . . basically, we try every classic web technique, but who are we kidding? Unless we dig out a zero-day on the official Outlook Web App 2019, we're not getting in.

NOTE *As fate would have it, at the time I was writing this close to 30,000 companies were compromised due to a flood of Microsoft Exchange vulnerabilities that granted a full working shell without any credentials—a clean remote code execution. You can see the exploit at* https://www.exploit-db.com/exploits/49637/*; for further reading see* https://bit.ly/3CgHY4s.

Next, we try scanning with Nmap in an attempt to find reachable services on the whole IP range owned by G&S Trust. The `-sV` flag displays the service's version, while `-p-` scans all 65,535 ports on each machine:

```
root@FrontLine:~# nmap -p- -sV 182.239.127.137-145
Starting Nmap 7.01 ( https://nmap.org )
Nmap scan report for 182.239.127.139
```

```
Host is up (0.023s latency).
Not shown: 65535 filtered ports

Nmap scan report for  182.239.127.139
Host is up (0.023s latency).
Not shown: 65534 filtered ports
PORT    STATE  SERVICE
443/tcp open   https
--snip--
```

Alas, this returns no significant results. Let's take a moment to recap. We are targeting a small company with maybe 15 or 25 employees, 7 of whom we can identify on the net. It has one webmail interface on the internet, four websites hosted by third parties, and nine allocated IP addresses, hardly exposing any services.

Have I bummed you out yet, or should I keep going?

Finding the Weak Links

When facing a target so small that even Google has trouble indexing its websites, you should always pause for a few seconds and think about the big picture. G&S Trust is not an island lost in the big blue sea that is the internet. The company must have multiple interactions with different vendors, business partners, and contractors to function properly in today's economic world. It's not called the World Wide Web for nothing.

So yes, G&S Trust might be immune to most of our reconnaissance probes because it has made it its mission and business model to stay off the grid—but what about its business partners? I'm not suggesting that we hack Microsoft or Apple to get inside this small company, no matter how many billions of dollars it buries in tax shelters. But surely it must have weaker and more exposed partners that we can infiltrate and use as a trampoline to bounce onto the internal network?

We go back again to our reconnaissance phase and scrape together every piece of news about G&S Trust that might disclose business partners or technologies it uses. We're looking for information on HR software, recent mergers and acquisitions, accounting software, and senior partners' backgrounds.

We won't go into all of them here, but you can find a plethora of open source intelligence (OSINT) tools at *https://github.com/jivoi/awesome-osint/*, from specialty search engines like BizNar that compile information on a given company, to document search tools such as *Sopdf.com*, to people and social media search engines. The availability of these tools almost makes it easy to forget how tedious this first step of compiling information really is.

Companies often unknowingly divulge a significant amount of information about their internal gears via various means—probably more than intended. Take job descriptions, for instance. We gain a trove of data simply by looking at an old IT support job listing G&S Trust posted on *Monster.com* (Figure 2-4).

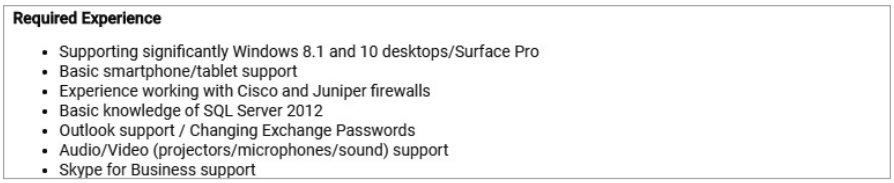


Figure 2-4: A G&S Trust job description

From this alone, we are able to figure out that at the time the job opening was listed G&S Trust used Windows 8.1 and 10 computers, worked mainly with SQL Server 2012 databases, and had installed Juniper and Cisco firewalls. While this might not help us get in directly (unless we use some form of social engineering), it might provide valuable insight once inside the network.

Since G&S Trust has offices in Hong Kong, we go to the official Hong Kong company registry website, *https://www.cr.gov.hk/*, and search for official filings. We find out that the company filed for a name change two years ago; it used to be called "GST Offshore Limited." Armed with this new information, we head back to our company research websites (*biznar.com*, *dnb.com*, *opencorporates.com*, and others) to hopefully unearth something useful.

On *dnb.com*, we find the names of three directors not currently listed on the G&S Trust website. We hunt them down on LinkedIn. All three profiles mention some obscure skills that seem to be related to the corporate finance world (Figure 2-5).

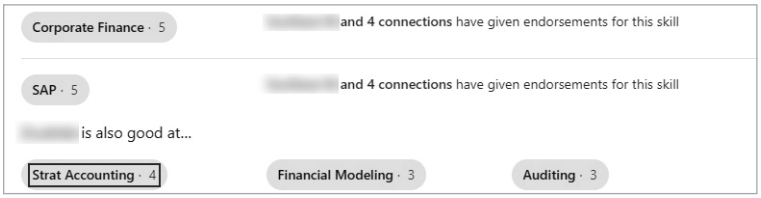


Figure 2-5: LinkedIn profiles revealing some G&S Trust inner workings

A quick Google search on the "Strat Accounting" skill reveals that this is in fact a software product owned and maintained by Strat Jumbo Inc., a multi-national development company.

Looking further into Strat Jumbo, we learn that it specializes in developing financial products, ranging from SWIFT connectors that plug into the banking network to workstation tools like Strat Accounting, which is apparently used by the financial employees at G&S Trust. Do you smell that? That's the scent of a very tempting yet dangerous idea taking shape somewhere in the darkness of our minds.

Since G&S Trust is so tightly locked down—from the outside, at least—how about targeting the much larger fish, Strat Jumbo, whose business we don't care about but which could perhaps give us free access to G&S Trust's network?

Of course, it's unlikely that Strat Jumbo is directly connected to G&S Trust's internal network. It's a software development company, not a strategic business partner. The more likely scenario is that we infiltrate Strat Jumbo's corporate network, locate Strat Accounting's code repository, and then plant a backdoor that gets triggered the next time G&S Trust updates its accounting software on the employee workstations. To execute this strategy, we'll have to carefully design a backdoor to only trigger in G&S Trust's environment. The last thing we want is to infect half the planet, causing mayhem and fury around the world (NotPetya, anyone?).

You may be wondering how this technique is different from phishing, which most of the time tends to yield the same result: executing payloads like reverse shells on a sizable number of workstations.

One word: trust.

Instead of shipping the payload in an email that goes through 36 security hoops before landing on the user's workstation, our payload is delivered by a trusted and verifiable emissary, Strat Jumbo.

If the antivirus software flags our email attachment, an investigation is usually quickly instigated; security analysts are called, and sometimes the issue escalates to the chief information security officer (CISO). On the other hand, if the antivirus flags the accounting software that has been used for the last 10 years, it takes just one call to the IT admin team and the antivirus is either disabled or adjusted to spare the defiant software. How's that for preferential treatment?

Resources

- Link to buy a USB Rubber Ducky: *https://hakshop.com/products/usb-rubber-ducky-deluxe/*
- Link to the Microsoft Exchange exploit: *https://www.exploit-db.com/exploits/49637/*
- Further reading on the Microsoft Exchange vulnerability: *https://www.welivesecurity.com/2021/03/10/exchange-servers-under-siege-10-apt-groups/*
- A collection of OSINT material and resources: *https://github.com/jivoi/awesome-osint/*
- A description of the NotPetya (aka Nyetna) attack: *http://blog.talosintelligence.com/2017/07/the-medoc-connection.html*