Two hip-hop artists are caught in a musical feud that extends into cyberspace. One artist's oversharing in his lyrics makes him a target, leading the rival label to hire a hacker to exploit vulnerabilities. As a security analyst for OWL Records, your role involves investigating these cyber activities by using data queries to uncover phishing attempts and unauthorized access.

Introduction

Alright, I'll state the obvious: rappers are bad at OSPEC. Think about it. The rap industry relies heavily on street cred, flashiness, and representing where you're from and what you've done. So, they often end up telling people things they shouldn't. As we will find out in the following KC7 threat hunting exercise, this makes them vulnerable.

In this scenario, two hip-hop artists, Dwake and Present, are in the midst of a musical feud. Following long-stewing tensions between the artists, they have begun taking jabs at each other through their music. Dwake, who is signed with OWL Records, was the first to strike. His newest song was intended to insult his arch-nemesis, Present, who is signed with Dollar Currency Records. However, he made a crucial mistake in his verse that took the feud in a different direction.

No, not in the direction you think; everyone is still alive.

Anyways, the role we play is that of a security analyst for OWL Records. Our job is to keep the company's information safe so the artists don't get exposed during this ongoing feud.

Dwake Drops His Verse

Dwake pulled no punches. After dropping this verse, he had everyone on social media talking and laughing at Present.

🎶 Yo, Present, you don't know where I'm from,
Got the Washington name from my mom's side, son.
It makes sense why they call you present
Cause you're so easy to beat, its pretty much a gift

Used to play with little Fluffy, now I'm runnin' with the wolves,
You say you're on top, but I'm breakin' all the rules.
I'm on that next level, you're stuck in the past,
with those weak beats you won't last.
🎶

And just like that, the beef escalated. I'm sure Dwake telling everyone certain details about his personal life won't have far-reaching consequences.

Now, understandably, Present was not pleased. In a fit of anger, Present asked his label, Dollar Currency Records (DCR), to dig up some dirt on Dwake that he could use to retaliate. Our "homeboy" (who works in the cyber underground) gave us a tip that we might see nefarious cyber activity as a result. For the high price of $20, he recounted a rumor he heard that DCR had hired a hacker who used the IP 18.66.52[.]227 to poke around our company's website in early April.

Now that we have a starting point, we can begin digging into the company's data to find clues that will help us solve the mysteries at hand. We'll manipulate our data using KQL (Kusto Query Language) queries. Time to pivot!

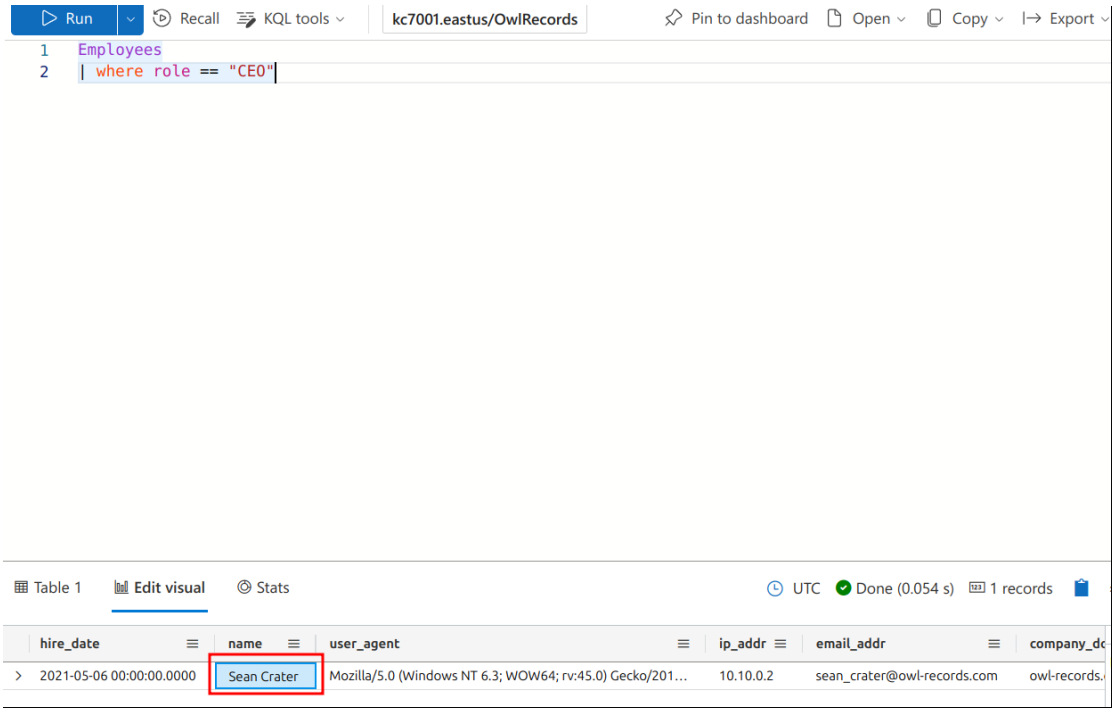
Initial Investigation

We are going to be manipulating the OwlRecords database, which has the following tables.

Available Tables

Table Name	Description
AuthenticationEvents	Records successful and failed logins to devices on the company network. This includes logins to the company's mail server.
Email	Records emails sent and received by employees.
Employees	Contains information about the company's employees.
FileCreationEvents	Records files stored on employee's devices.
InboundNetworkEvents	Records inbound network events including browsing activity from the Internet to devices within the company network.
OutboundNetworkEvents	Records outbound network events including browsing activity from within the company network out to the Internet.
PassiveDns (External)	Records IP-domain resolutions.
ProcessEvents	Records processes created on employee's devices.
SecurityAlerts	Records security alerts from an employee's device or the company's email security system.

The first question KC7 asks is, "What is the name of the OWL Records CEO?" This can be answered with a simple query.



What is the name of the OWL Records CEO?

✅ Sean Crater

The next question is how many results we get back from running the following query:

```
1  InboundNetworkEvents
2  | where timestamp between (datetime("2024-04-10T00:00:00") .. datetime("2024-04-11T00:00:00"))
3  | where src_ip has "18.66.52.227"
4
```

The answer lies in the upper right corner!

Sidenote: These are some interesting results. Somebody's curious. We definitely should dig into these more.

timestamp	method	src_ip	user_agent	url	status_co
2024-04-10 00:00:00.0000	GET	18.66.52.227	Mozilla/5.0 (compatible; MSIE 6....	https://owl-records.com/search=whats+Dwake%27s+email...	200
2024-04-10 10:50:35.0000	GET	18.66.52.227	Mozilla/5.0 (compatible; MSIE 6....	https://owl-records.com/search=OWL+Records+rapper+co...	200
2024-04-10 10:51:10.0000	GET	18.66.52.227	Mozilla/5.0 (compatible; MSIE 6....	https://owl-records.com/search=OWL+Records+artists+con...	200
2024-04-10 10:51:34.0000	GET	18.66.52.227	Mozilla/5.0 (compatible; MSIE 6....	https://owl-records.com/search=how+do+i+email+Dwake...	200
2024-04-10 10:51:55.0000	GET	18.66.52.227	Mozilla/5.0 (compatible; MSIE 6....	https://owl-records.com/search=Dwake+booking+info+pls	200
2024-04-10 10:52:00.0000	GET	18.66.52.227	Mozilla/5.0 (compatible; MSIE 6....	https://owl-records.com/search=OWL+Records+artist+ema...	200
2024-04-10 10:52:02.0000	GET	18.66.52.227	Mozilla/5.0 (compatible; MSIE 6....	https://owl-records.com/search=can+i+book+Dwake+for+...	200
2024-04-10 10:52:38.0000	GET	18.66.52.227	Mozilla/5.0 (compatible; MSIE 6....	https://owl-records.com/search=why+is+Dwake+music+m...	200
2024-04-10 10:53:07.0000	GET	18.66.52.227	Mozilla/5.0 (compatible; MSIE 6....	https://owl-records.com/artists/Dwake/	200
2024-04-10 10:53:42.0000	GET	18.66.52.227	Mozilla/5.0 (compatible; MSIE 6....	https://owl-records.com/artists/email_contacts/	200
2024-04-10 10:54:15.0000	GET	18.66.52.227	Mozilla/5.0 (compatible; MSIE 6....	https://owl-records.com/artists/booking_info/	200
2024-04-10 10:55:08.0000	GET	18.66.52.227	Mozilla/5.0 (compatible; MSIE 6....	https://owl-records.com/events/OWLRecords_concerts_2024/	200
2024-04-10 10:55:56.0000	GET	18.66.52.227	Mozilla/5.0 (compatible; MSIE 6....	https://owl-records.com/marketing/artist_profiles/	200
2024-04-10 10:56:30.0000	GET	18.66.52.227	Mozilla/5.0 (compatible; MSIE 6....	https://owl-records.com/marketing/press_releases/	200
2024-04-10 10:57:03.0000	GET	18.66.52.227	Mozilla/5.0 (compatible; MSIE 6....	https://owl-records.com/legal/contact_us/	200

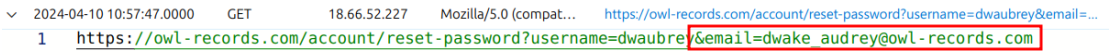
The results we are looking at represent someone browsing and searching for information on OWL Records' website. This operator (the person browsing the website) was clearly looking to find information about various artists who work for OWL Records, especially Dwake. Thanks to our contact, it looks like we are on the right path. However, a question remains: What key piece of information were they looking to get for Dwake? The answer lies in that first result.

https://owl-records.com/search=whats+Dwake%27s+email+address%3F

Let's continue looking at the results from the previous query to answer the next question. The operator also expressed strong opinions about Dwake's music. What were they wondering?

https://owl-records.com/search=why+is+Dwake+music+much+soo+trasshhhh

When we continue to look at these logs, we discover that the operator discovered Dwake's email address at some point.



The operator then attempted to take over Dwake's account by resetting his password. We know this because of the reset-password parameter in the last url they accessed. When employees at OWL records need to reset their passwords, they must answer a set of challenge questions to prove who they are. These are the challenge questions offered on the OWL records website:

1.

What is your mother's maiden name?

2.

What street did you grow up on as a child?

3.

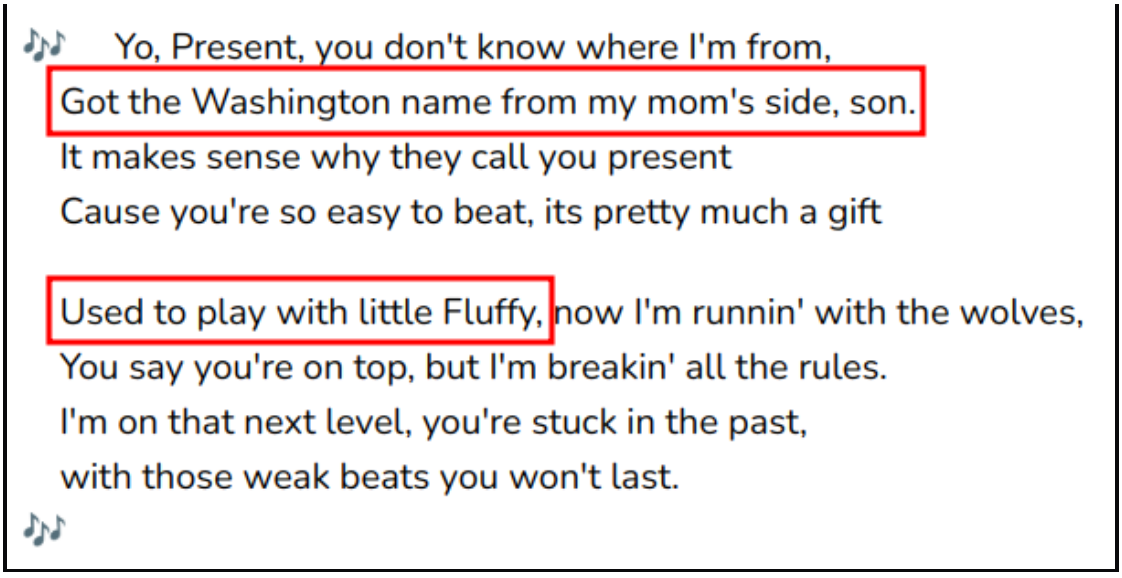
What is your childhood pet's name?

4.

What is the color of your first car?

5.

You see what I'm seeing, right? Based on the questions, we can conclude that our favorite rap artist accidentally self-snitched in his last verse.



The operator used these bits of information to reset Dwake's password. By using a certain query, we can actually see the adversary doing this.

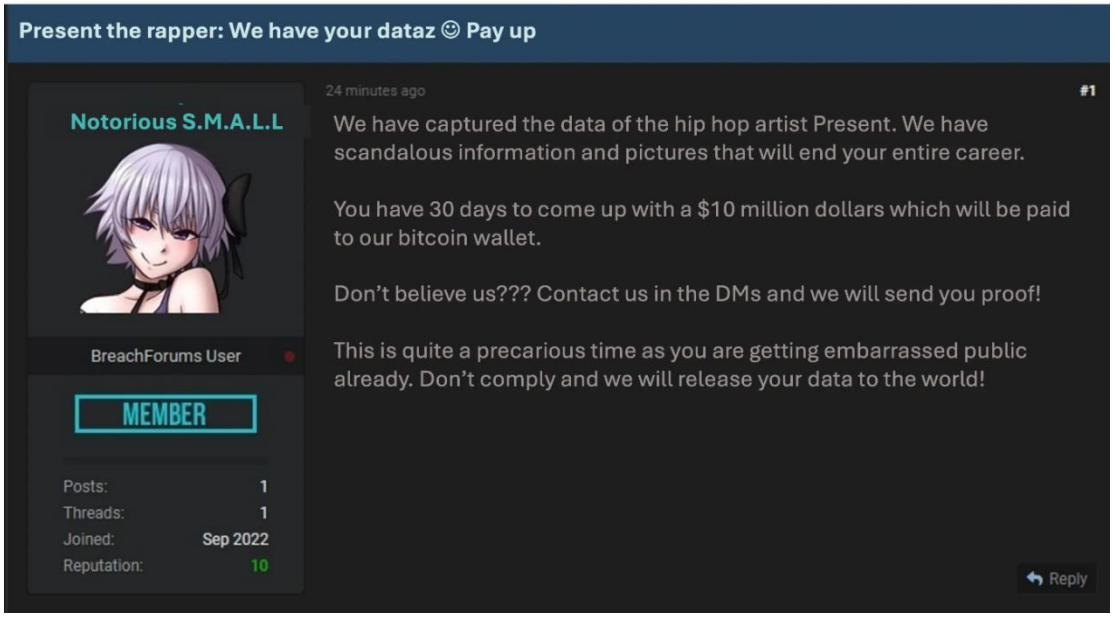
```
InboundNetworkEvents
| where timestamp between (datetime("2024-04-10T00:00:00") .. datetime("2024-04-11T00:00:00"))
| where url has_any("washington", "fluffy")
| where src_ip has "18.66.52.227"
```


```
"timestamp": 2024-04-10T13:02:47Z,
"method": GET,
"src_ip": 18.66.52.227,
"user_agent": Mozilla/5.0 (compatible; MSIE 6.0; Windows NT 6.2; Trident/5.0),
"url": https://owl-records.com/account/security-questions?question_1=mother's+maiden+name&answer_1=Washington&
  question_2=first+pet's+name&answer_2=Fluffy,
"status_code": 200
```


This worked in the hacker’s favor, and after taking over Dwake's email account, they were able to reset the password for Dwake's Instagram account as well. The following day, the adversaries posted an embarrassing image to Drake's Instagram.

The higher-ups at OWL records deliberated the event in a private meeting. In that meeting was Dwake himself, who was seething about what had taken place. After 6 hours of deliberation, the company declared it a settled issue. However interestingly enough, they never officially specified how they dealt with the situation.

The following day, a random hacker on the dark web threatened to release damaging information on Present the rapper, if he did not announce his retirement in the next 30 days.



OWL Records is totally not involved. □

We can't really prove that OWL Records management had anything to do with the dark web post (and honestly, we don't have any reason to try). But we know this could cause us more headaches if it's not sorted out. So, we hit up our contact in the cyber underground again, and he hinted that any payback might come through phishing. It's a bit of a vague clue, but we're pros, so we can definitely figure something out.

Hunting for Phish

We already know one thing about the bad guys! They used IP 18.66.52[.]227 in their operations. If we can track down a domain name linked to this IP, it might lead us to those phishing emails

We can look in the PassiveDNS table for ip <-> domain relationships to find the adversary's domain. Hunting for Phish

```
PassiveDns
| where ip == "18.66.52.227"
```

timestamp	ip	domain
2024-04-07 07:50:13.0000	18.66.52.227	betterlyrics4u.com

Woohoo! We've got a domain name! Things should be a lot easier now! Let's dive into the email logs to see if this domain was used. Before peering at the logs, we need to inspect the email table.

Let's check out the table layout and figure out which column probably has our domain. By using "take 10" to show just the top ten results, we can see that the link column has URLs. That's probably where we'll find our domain.

```
1  Email
2  | take 10
```

timestamp	sender	reply_to	recipient	subject	verdict	link
2024-04-01 07:44:...	larry_russell@beat...	larry_russell@beat...	greg_wells@owl-re...	[EXTERNAL] RE:RE: Em...	CLEAN	https://drive.google.com/driv...
2024-04-01 07:57:...	chance_west@owl-...	chance_west@owl-...	lil_zee@owl-record...	FW: Also respect use w...		https://docs.google.com/spre...
2024-04-01 08:15:...	joseph_johnson@o...	joseph_johnson@o...	christine_baran@b...	The the conflict the the...		https://docs.google.com/spre...
2024-04-01 08:36:...	lauri_lawrence@ow...	lauri_lawrence@ow...	megannguyen@ya...	Rake conflict his grew a...		https://images.owl-records.co...
2024-04-01 08:55:...	samuel_manson@b...	samuel_manson@b...	benny_nixon@owl-...	[EXTERNAL] FW: New s...	CLEAN	https://kcfoundation.org/pub...

Now, we should look for the “betterlyrics4u[.]com” domain.

```
Email
| where link has "betterlyrics4u.com"
```

The query returns 13 results. Each row represents an email sent to someone at OWL records! We need to find the email address that sent most of the emails. Multiple emails sent from the same external domain to different company email addresses could be a sign of a targeted phishing campaign.

We can easily spot the outlier.

timestamp	sender	reply_to	recipient	subject	verdict
2024-04-10 14:45:...	ghostwritersanonymous@protonmail.com	ghostwritersanony...	ice_blake@owl-records.com	[EXTERNAL] Get FREE b...	CLEAN
2024-04-10 14:45:...	ghostwritersanonymous@protonmail.com	ghostwritersanony...	nikki_lane@owl-records.com	[EXTERNAL] Get FREE b...	CLEAN
2024-04-11 08:50:...	wemakebeatz@gmail.com	wemakebeatz@gm...	jay_knight@owl-records.com	[EXTERNAL] FW: Get FR...	BLOCKED
2024-04-11 08:50:...	wemakebeatz@gmail.com	wemakebeatz@gm...	jay_knight@owl-records.com	[EXTERNAL] FW: Get FR...	BLOCKED
2024-04-12 13:37:...	ghostwritersanonymous@protonmail.com	ghostwritersanony...	lil_zee@owl-records.com	[EXTERNAL] Need a gh...	CLEAN
2024-04-12 13:37:...	ghostwritersanonymous@protonmail.com	ghostwritersanony...	jay_knight@owl-records.com	[EXTERNAL] Need a gh...	CLEAN
2024-04-15 11:07:...	ghostwritersanonymous@protonmail.com	ghostwritersanony...	dwake_audrey@owl-records.com	[EXTERNAL] RE: Need a...	CLEAN
2024-04-15 11:07:...	ghostwritersanonymous@protonmail.com	ghostwritersanony...	snoop_thompson@owl-records.com	[EXTERNAL] RE: Need a...	CLEAN
2024-04-17 09:46:...	ghostwritersanonymous@protonmail.com	ghostwritersanony...	justin_cole@owl-records.com	[EXTERNAL] RE:RE: Nee...	CLEAN
2024-04-17 09:46:...	ghostwritersanonymous@protonmail.com	ghostwritersanony...	chance_west@owl-records.com	[EXTERNAL] RE:RE: Nee...	CLEAN
2024-04-19 13:45:...	ghostwritersanonymous@protonmail.com	ghostwritersanony...	drake_hill@owl-records.com	[EXTERNAL] Need a gh...	CLEAN
2024-04-19 13:45:...	ghostwritersanonymous@protonmail.com	ghostwritersanony...	drake_hill@owl-records.com	[EXTERNAL] Need a gh...	CLEAN
2024-04-25 14:36:...	ghostwritersanonymous@protonmail.com	ghostwritersanony...	logic_white@owl-records.com	[EXTERNAL] Get FREE b...	CLEAN

However, we also see a second email address of interest. Luckily emails from this one were blocked.

What was the other email address used to send these phishing emails?

wemakebeatz@gmail.com

Okay, we managed to pivot from the domain to find the email addresses for the phishing campaign. To further support our position that this is indeed a targeted effort, we should find out which job role was targeted the most. We need to inspect the email table by assigning it to a variable, filter out unique recipients, and cross-reference the results with the email_addr column in the Employees table.

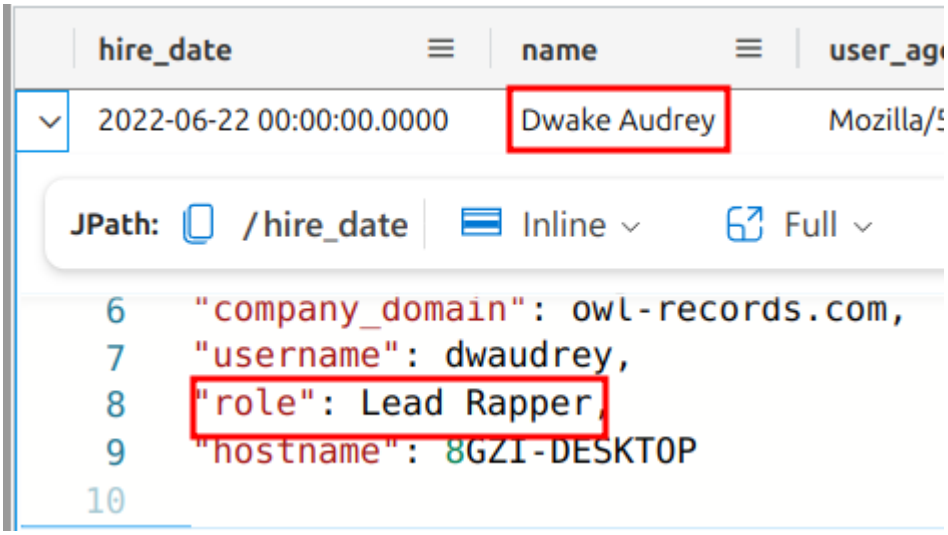
```
let _targets = Email
| where link has "betterlyrics4u.com"
| distinct recipient;
Employees
| where email_addr in (_targets)
| summarize count() by role
```

It seems our attacker only wanted to target the rappers.

role	count_
Rapper	9
Lead Rapper	1

Interesting! There is only one result for the lead rapper. I wonder who it could be.

```
let _targets = Email
| where link has "betterlyrics4u.com"
| distinct recipient;
Employees
| where email_addr in (_targets)
| where role == "Lead Rapper"
```



Oh. Yeah, that makes sense, I guess. Personally, I think Snoop is better, but that's just me. □□♂

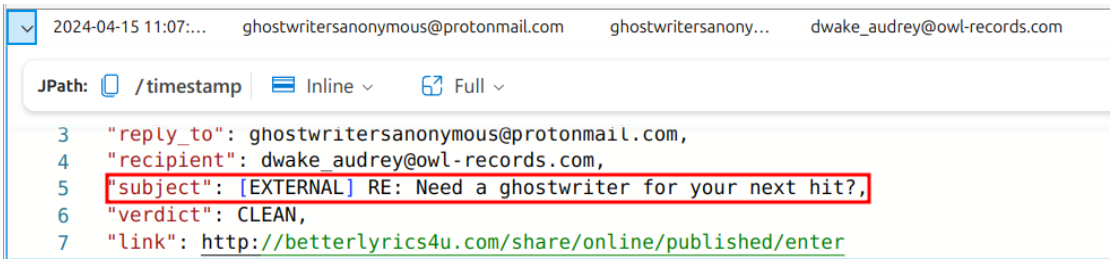
Before we pivot off this, we should note Dwake's IP address. If the phishing attack was successful, this will help later if we need to check the authentication logs.

What is Dwake's IP address?

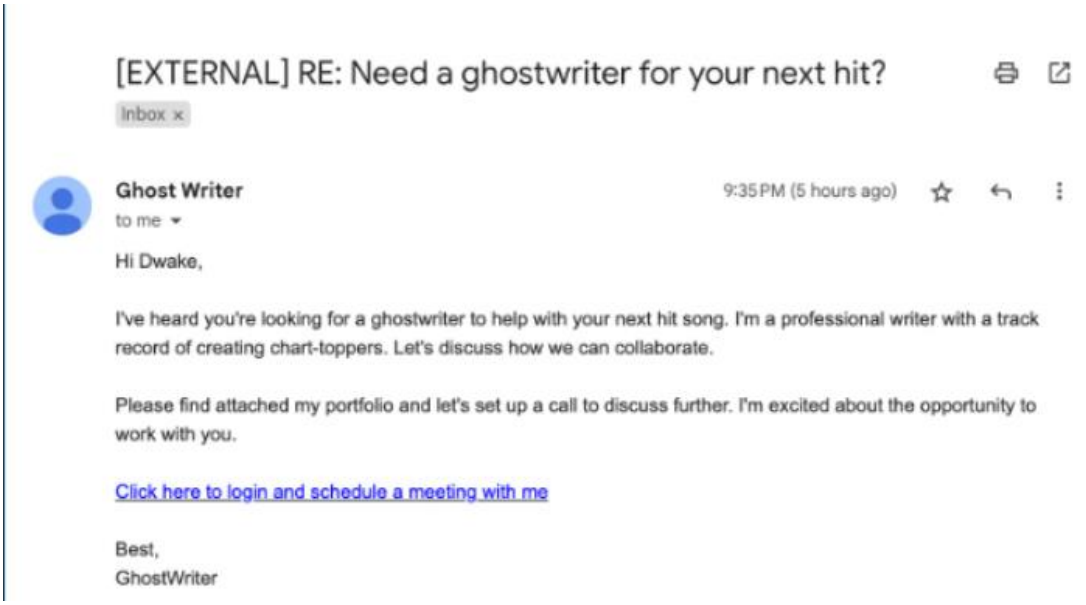
10.10.0.5

The next question we have to answer is: What is the subject of the email sent to Dwake? Since this is a spear-phishing attack, it has to be interesting enough for him to open.

```
1  Email
2  | where link has "betterlyrics4u.com"
3  | where recipient contains "dwake"
```



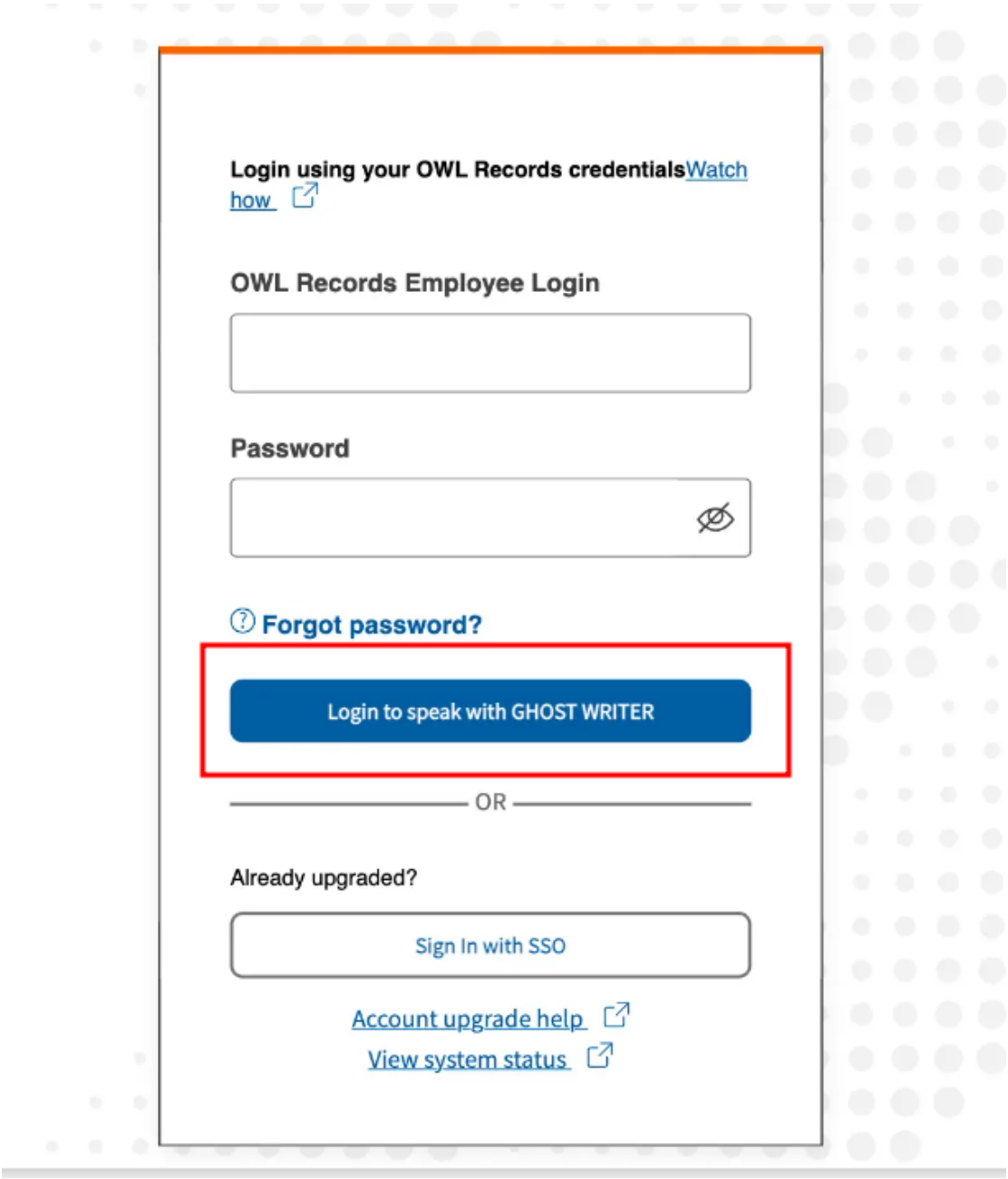
Besides noticing that Dwake probably doesn't write his own bars, we get quite a bit of information from this result. For one, this email got through the spam filter and was dubbed "CLEAN." We also see the link the attacker wants him to follow. Let's inspect the actual contents of the email to get a better idea of what we're working with.



If Dwake clicked the link, we would see the data in the OutboundNetworkEvents table. By examining the data here, we can confirm there was a GET request for the malicious URL and the timeframe in which it happened.

```
1  OutboundNetworkEvents
2  | where url == "http://betterlyrics4u.com/share/online/published/enter"
3  | where src_ip == "10.10.0.5"
```

After Dwake clicked on the link in the email, he was presented with this phishing page:



Most likely, he entered his username and password, and the credentials were sent to a remote server controlled by the attacker. Now that they have his credentials, the attackers would need to verify them by logging into Dwake's account. As I mentioned earlier, we can check the AuthenticationEvents table to see if the attackers were able to log in to Dwake's account. We'll need to use the attacker's IP address for this.

```
1  AuthenticationEvents
2  | where username == "dwaudrey"
3  | where src_ip == "18.66.52.227"
4
```

It seems the attacker was able to successfully login.

```
"timestamp": 2024-04-15T13:03:12Z,
"hostname": MAIL-SERVER01,
"src_ip": 18.66.52.227,
"user_agent": Mozilla/5.0 (compatible; MSIE 6.0; Windows NT 6.2; Trident/5.0),
"username": dwaudrey,
```


Now that the adversaries have logged into Dwake's account, they will want to look for important information to steal. Since we already know the adversary's IP address, we can check for InboundNetwork events during that timeframe and hunt for activity against Dwake's account.

```
InboundNetworkEvents
| where timestamp between (datetime("2024-04-12T00:00:00") .. datetime("2024-05-01T00:00:00"))
| where url has "dwaudrey"
| where src_ip has "18.66.52.227"
```

We get ten records with this query and some suspicious URL parameters.

Table 1 | Add visual | Stats | Search | UTC | Done (0.375 s) | 10 records

timestamp	method	src_ip	user_agent	url	status_coc
2024-04-16 09:49:32.0000	GET	18.66.52.227	Mozilla/5.0 (compat...	https://owl-records.com/mail?user=dwaudrey/search?query=scandal	200
2024-04-16 09:50:04.0000	GET	18.66.52.227	Mozilla/5.0 (compat...	https://owl-records.com/mail?user=dwaudrey/search?query=confidential	200
2024-04-16 09:50:13.0000	GET	18.66.52.227	Mozilla/5.0 (compat...	https://owl-records.com/mail?user=dwaudrey/search?query=leak	200
2024-04-16 09:50:23.0000	GET	18.66.52.227	Mozilla/5.0 (compat...	https://owl-records.com/mail?user=dwaudrey/search?query=secret	200
2024-04-16 09:50:27.0000	GET	18.66.52.227	Mozilla/5.0 (compat...	https://owl-records.com/mail?user=dwaudrey/search?query=sensitive	200
2024-04-16 09:51:12.0000	GET	18.66.52.227	Mozilla/5.0 (compat...	https://owl-records.com/mail?user=dwaudrey/drafts/	200
2024-04-16 09:51:59.0000	GET	18.66.52.227	Mozilla/5.0 (compat...	https://owl-records.com/mail?user=dwaudrey/internal_communications/meetin...	200
2024-04-16 09:52:09.0000	GET	18.66.52.227	Mozilla/5.0 (compat...	https://owl-records.com/mail?user=dwaudrey/internal_communications/private_...	200
2024-04-16 09:52:41.0000	GET	18.66.52.227	Mozilla/5.0 (compat...	https://owl-records.com/mail?user=dwaudrey/internal_communications/employ...	200
2024-04-30 14:23:02.0000	GET	18.66.52.227	Mozilla/5.0 (compat...	https://owl-records.com/mail/readmail?user=dwaudrey%40owl-records.com&m...	200

It looks like the attacker accomplished their goal. We see evidence of data exfiltration using a zip file.

2024-04-30 14:23:02.0000 GET 18.66.52.227 Mozilla/5.0 (compat... https://owl-records.com/mail/readmail?user=dwaudrey%40owl-records.com&...

```
https://owl-records.com/mail/readmail?user=dwaudrey%40owl-records.com&mailbox_folder=Drafts&download=true&
output=DwakesDirtySecrets.zip
```

Conclusion

Dwake and Present have a lot of dirt on each other now. After some back-and-forth through some middlemen, they decided to call a truce. Luckily, this didn’t escalate beyond cyberspace. Everything's chill in the Rap World for now!

This KC7 exercise demonstrated the entire cyber attack process—from reconnaissance and phishing to account compromise and data theft—in an entertaining way. It highlighted the importance of detailed data analysis, the effectiveness of KQL for threat hunting, and the value of communication in engaging both technical and non-technical audiences. It also showed the importance of training employees on operational security (OPSEC), especially in industries where reputation and persona are public-facing.

If personal info gets leaked or shared too much, it can be used in social engineering attacks. I'm definitely going to keep using KC7 as a hands-on learning tool. This

exercise was a big reminder that cybersecurity needs to tackle both the technical and human sides to really work well.