باسمه‌تعالی

آسیب‌پذیری portmapper

و نحوه امن‌سازی آن

مقدمه

برای بررسی آسیب‌پذیری portmapper باید ابتدا مفهوم RPC (Remote Procedure Call) بیان شود.

معرفی RPC

RPC پروتکلی است که توسط آن یک برنامه می‌تواند از یک برنامه موجود در یک سیستم دیگر که در شبکه قرار دارد، درخواست سرویس نماید بدون آن‌که نیاز به دانستن جزئیات شبکه داشته باشد. RPC از مدل کلاینت-سرور استفاده می‌کند. برنامه‌ای که درخواست سرویس می‌دهد، کلاینت و برنامه‌ای که در طرف دیگر به این درخواست پاسخ می‌دهد و سرویس مورد نظر را فراهم می‌کند، سرور می‌باشد. RPC همانند یک روند فراخوانی معمولی، عملیاتی سنکرون می‌باشد بدین معنی که برنامه‌ای که درخواست سرویس می‌دهد باید تا بازگشت نتیجه از سرور راه دور، به صورت معطل باقی بماند. البته استفاده از پردازنده‌هایی که یک فضای آدرس مشابه از حافظه را به اشتراک می‌گذارند، موجب می‌شود که چندین RPC بتوانند به طور همزمان اجرا شوند.

بر اساس مدل OSI، RPC یک پروتکل لایه کاربرد (و نه پروتکل لایه انتقال) می‌باشد. البته RPC از ویژگی‌های ارتباطی موجود در لایه انتقال نیز استفاده می‌کند. با استفاده از RPC می‌توان برنامه‌های کاربردی که شامل چندین برنامه توزیع‌شده در سطح شبکه می‌باشند را به آسانی تولید کرد.

در سرورهایی که دارای سیستم‌عامل‌های مبتنی بر Unix می‌باشند، برنامه‌هایی مانند lock manager، NFS daemons و license manager ها از RPC استفاده می‌کنند. همچنین اکسپلویت‌های بسیاری برای آن وجود دارد و هر روز هم به تعداد آن‌ها افزوده می‌شود. اولین گام برای بهره‌برداری و سوء استفاده از یک سرویس آن است که ابتدا تشخیص داده شود که سرویس بر روی سیستم هدف در حال اجرا می‌باشد. در این مرحله portmapper و rpcbind و پورت شناخته شده 111 نیز وارد ماجرا می‌شوند.

Portmapper و rpcbind

portmapper یک برنامه RPC و شماره نسخه آن را به یک شماره پورت خاص در لایه انتقال نگاشت می‌کند. portmapper برای ثبت یک RPC از شناسه‌هایی مانند شماره سرویس RPC، شماره نسخه، پروتکل مورد استفاده و پورت TCP یا UDP استفاده می‌کند. portmapper امکان نگاشت برنامه‌های راه دور را به صورت پویا

فراهم می‌کند. همچنین portmapper همیشه بر روی پورت 111 پروتکل TCP یا UDP اجرا می‌شود. برنامه‌هایی که در مکانیزم RPC به عنوان سرور عمل می‌کنند، از پورت‌های ناشناخته و موقتی استفاده می‌کنند و بنابراین کلاینت‌ها برای برقراری ارتباط با این برنامه‌ها نیاز به دانستن یک شماره پورت شناخته شده از طرف آن‌ها دارند تا بتوانند ارتباط برقرار کنند. بنابراین زمانی که یک کلاینت بخواهد به سرویسی دسترسی داشته باشد، باید ابتدا با portmapper ارتباط برقرار کند و پس از آن portmapper شماره پورت مورد نظر برای دسترسی کلاینت به سرویس مورد درخواست را در اختیارش قرار می‌دهد. بنابراین دسترسی به پورت 111 به برنامه‌هایی که به عنوان کلاینت عمل می‌کنند اجازه می‌دهد تا بتوانند پورت‌های ناشناخته‌ای که توسط سرور تعریف شده است را تشخیص دهند. اگر portmapper وجود نداشته باشد یا در دسترس نباشد، درخواست کلاینت رد می‌شود.

متاسفانه RPC امنیت بسیار پائینی دارد. ضمناً به این دلیل که سرویس‌های مبتنی بر RPC برای برقراری ارتباط نیاز به rpcbind دارند، باید قبل از شروع سرویس‌های مورد نظر، ابتدا rpcbind در دسترس باشد.

زمانی که یک کلاینت از طریق مکانیزم RPC یک شماره برنامه را فراخوانی می‌کند، ابتدا به منظور تشخیص آدرسی که باید درخواست RPC را به آن ارسال کند، به rpcbind متصل می‌شود. اگر پورت 111 فعال باشد، لیستی از تمام سرویس‌های فعال مهیا بوده و به این ترتیب می‌تواند آدرسی که کلاینت باید به آن متصل شود را در اختیارش قرار دهد. البته در بعضی از نسخه‌های Unix و Solaris، rpcbind نه تنها به پورت 111 از نوع TCP و UDP گوش می‌کند، بلکه به پورت‌های UDP بزرگتر از 32770 نیز گوش می‌دهد. شماره‌های دقیق پورت‌ها به ساختار سیستم‌عامل و نسخه آن بستگی دارد. بنابراین تجهیزات فیلترینگ و ACL های استفاده شده در روترها و فایروال‌ها که برای عدم دسترسی به rpcbind یا portmapper بر روی شماره پورت 111 تنظیم شده‌اند، می‌توانند با ارسال یک درخواست UDP به rpcbind و به شماره پورت‌های بالاتر از 32770 دور زده شوند. کاربران غیرمجاز با سوء استفاده از این آسیب‌پذیری می‌توانند اطلاعات RPC یک سیستم راه دور را به‌دست آورند (حتی در صورتی که پورت 111 مسدود شده باشد).

با توجه به اطلاعات RPC که از طریق پورت 111 به‌دست می‌آید، می‌توان فهمید که چه سرویس‌هایی در حال اجرا می‌باشند. در این رابطه تعداد زیادی آسیب‌پذیری وجود دارد که اکسپلویت‌های مرتبط با آن‌ها هم قابل

دسترسی است. بر روی سیستم‌هایی که به طور کامل از آن‌ها محافظت نمی‌شود و portmapper بر روی آن‌ها درحال اجراست، با اجرای یک دستور ساده rpcinfo می‌توان لیستی از تمام سرویس‌هایی که در حال اجرا می‌باشند را به‌دست آورد.

نحوه امن‌سازی portmapper

نحوه احراز اصالت در portmapper ضعیف می‌باشد و به دلیل اینکه portmapper می‌تواند تعداد زیادی از شماره پورت‌ها را به سرویس‌های تحت کنترلش اختصاص دهد، امن‌سازی آن مشکل می‌باشد. به هر حال، اگر سرویس RPC در حال اجرا می‌باشد می‌توان مراحل زیر را انجام داد:

امن‌سازی portmapper با استفاده از TCP Wrappers

با استفاده از TCP Wrappers می‌توان تعیین کرد که کدام شبکه‌ها و یا سیستم‌ها مجاز به دسترسی به سرویس portmap می‌باشند. بنابراین هنگامی که قصد محدود کردن دسترسی به سرویس وجود داشته باشد، باید فقط از آدرس‌های IP استفاده کرد و نباید از نام سیستم (hostname) در این حالت استفاده شود، چون نام‌ها می‌توانند با حملاتی مانند DNS poisoning و یا مشابه آن جعل شوند.

امن‌سازی portmapper با استفاده از IPTables

برای محدود کردن دسترسی به سرویس portmap به یک شبکه خاص، می‌توان rule های زیر را به iptables اضافه کرد. در این دو rule فقط به درخواست‌های اتصال TCP برای سرویس portmap از طرف شبکه‌های 192.168.0.0/24 و localhost اجازه داده شده و بقیه بسته‌ها drop می‌شوند:

```
iptables -A INPUT -p tcp -s! 192.168.0.0/24  --dport 111 -j DROP
iptables -A INPUT -p tcp -s 127.0.0.1  --dport 111 -j ACCEPT
```

برای محدود کردن ترافیک UDP نیز می‌توان از دستور زیر استفاده کرد:

```
iptables -A INPUT -p udp -s! 192.168.0.0/24  --dport 111 -j DROP
```