No es una impresora, es una máquina de voto

Ignacio E. Losiggio

29/09/17 (ekoparty)

$ whoami

Ignacio Losiggio (@memepowered)

- Dev. en Huayra GNU/Linux
- Estudio CS de la Computación (UBA-Exactas)
- Dormidor compulsivou

1. Es una impresora

¿Qué puede salir mal?

1. Es una impresora

¿Qué puede salir mal?

Podemos pensarlas como los primeros dispositivos IOT, si algo sabemos del interné de las cosas es que está todo roto (mal software, configuración nula, passwords por defecto, etc).

2. Caso de estudio: Ilusionistas informáticos

RELLENAR CON EL TRUCO DE MAGIA EN EL CONGRESO CON MICROIMPRESIONES

2. Caso de estudio: Ilusionistas informáticos

Si no asumimos malicia en el sistema ¿Cómo nos divertimos?

3. PostScript, programando hojas de papel

Las páginas de impresión se especifican en PostScript en muchas impresoras (incluso en muchas modernas por retrocompatibilidad)

Características:

- Stack-based

3. PostScript, programando hojas de papel

Las páginas de impresión se especifican en PostScript en muchas impresoras (incluso en muchas modernas por retrocompatibilidad)

Características:

- Stack-based
- Usado casi exclusivamente para impresión y administración de impresoras

3. PostScript, programando hojas de papel

Las páginas de impresión se especifican en PostScript en muchas impresoras (incluso en muchas modernas por retrocompatibilidad)

Características:

- Stack-based
- Usado casi exclusivamente para impresión y administración de impresoras
- **El estado se revierte finalizado el trabajo de impresión**

3. PostScript, programando hojas de papel

El verbo que necesitamos: `startjob`

Limpia todos los stacks (gráfico, operadores, ejecución, diccionarios, etc), restaura el estado inicial de la VM de PostScript y continúa la ejecución desde el siguiente verbo *del trabajo de impresión*.

3. PostScript, programando hojas de papel

El verbo que necesitamos: `startjob`

Limpia todos los stacks (gráfico, operadores, ejecución, diccionarios, etc), restaura el estado inicial de la VM de PostScript y continúa la ejecución desde el siguiente verbo *del trabajo de impresión*.

Se puede usar de dos formas posibles:

- `true <password> startjob`; Crea un nuevo trabajo de impresión, **los cambios al estado son persistentes**
- `false <password> startjob`: Crea un nuevo trabajo de impresión

3. PostScript, programando hojas de papel

Idea

Generar un trabajo de impresión que condicione los siguientes (deniegue ciertas boletas o formularios, imprima cosas inesperadas, **cuente boletas de papel**)

vot.ar.ps, tareas:

1. Detectar que la página imprimiéndose es una boleta
2. Cambiar a contexto persitente, contar el voto
3. Cambiar a contexto común, restaurar los gráficos anteriores e imprimir

4. Problemas para seguir investigando

- La implementación actual depende de que el trabajo de impresión sea identificable únicamente en tres lugares independientes
- PostScript es complejo y es posible que tenga errores
  ¿Podemos ejecutar algo por fuera de la VM? (Transmitir datos en tiempo real en impresoras con red)
- Hay otras formas de describir las páginas que son más modernas
- Nuestra impresora no tiene reloj :(

5. Conclusión

Las impresoras de oficina pueden ser un target útil si se conoce con buen detalle la estructura a atacar

6. Fuentes y recursos

- PostScript Language Reference Manual: https://www-cdf.fnal.gov/offline/PostScript/PLRM2.pdf
- Printer Exploitation Toolkit: https://github.com/RUB-NDS/PRET
- Hacking Printers: http://hacking-printers.net/wiki/index.php/Main_Page
- vot.ar.ps: https://github.com/iglosiggio/vot.ar.ps