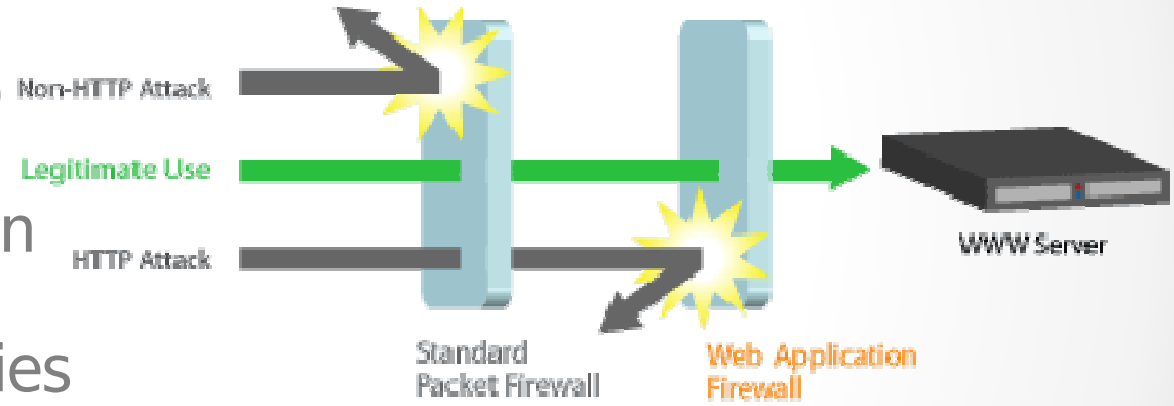
NAXSI

OWASP NAXSI Project

Ricardo Supo Picon
ricardo@limasoft.com

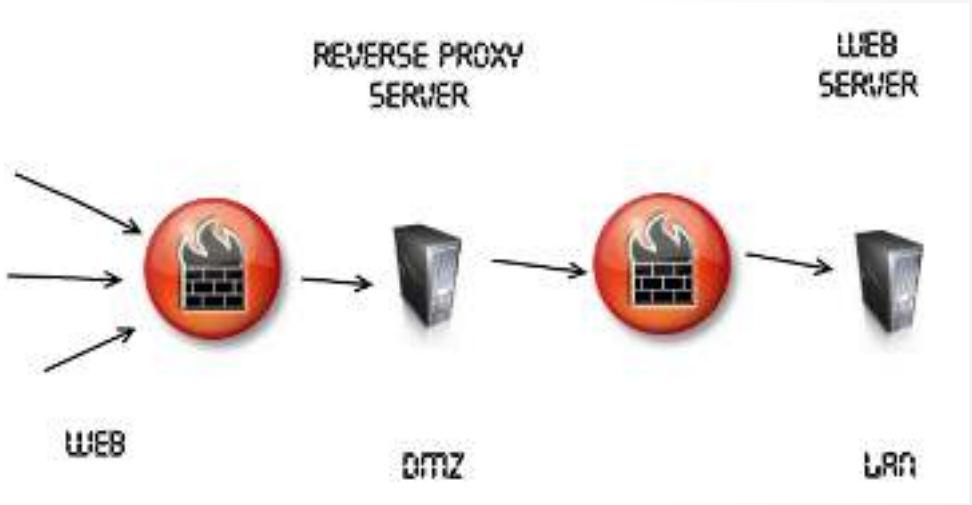
Web Application Firewal

- Protege los servidores Web de trafico malicioso y bloquea los intentos de comprometer el sistema.
- Previene los ataques web cross site scripting, inyección SQL, navegación contundente, el envenenamiento de cookies y la entrada no valida.
- Hardware, plugin de servidor, o un filtro que se aplica un conjunto de reglas a una conversacion HTTP.



Proxy Reverso

- Todo el tráfico procedente de Internet y con destino en alguno de esos servidores web es recibido por el servidor proxy.
- Seguridad: el servidor proxy es una capa adicional
- Cifrado / Aceleración SSL
- Distribución de Carga
- Caché de contenido estático



Nginx Anti Xss Sql Injection

- Proyecto OWASP
- Modulo Open Source de WAF para Nginx.
- De alto rendimiento, mantenimiento reglas bajas,
- Protege contra inyecciones SQL, Cross Site Scripting, Cross Site Request Forgery, inclusiones de archivos locales y remotos.

¿Cómo funciona?

- Modo aprendizaje.
- Genera lista blanca.
- Reduce falsos positivos.
- No se basa en firmas predefinidas.
- Reduces ataques ofuscados

¿Cómo Instalar?

apt-get instal nginx-naxsi

yum install nginx-naxsi

¿Cómo Instalar?

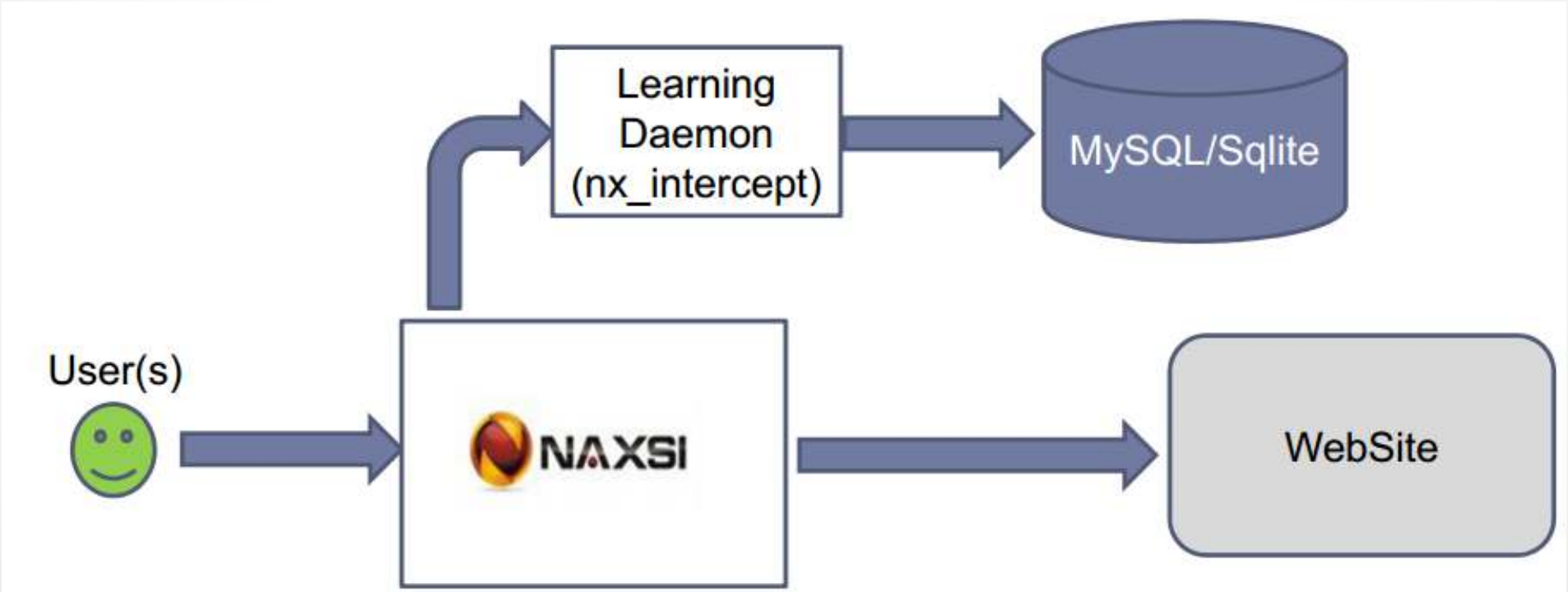
apt-get instal nginx-naxsi

yum install nginx-naxsi

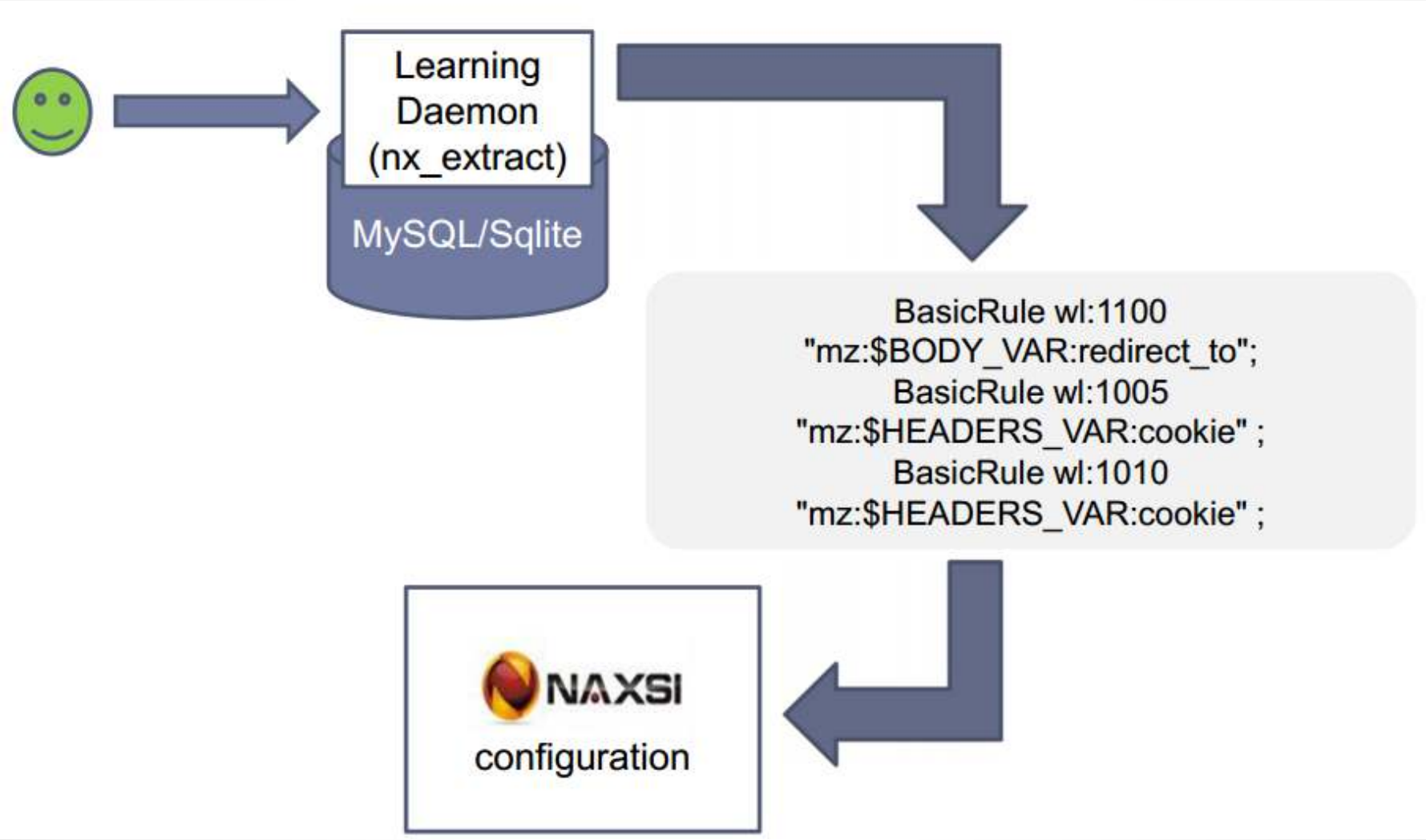
Learning Mode

- Utiliza script en python script que parsea los logs de nginx.
- Usa modo standalone python HTTP daemon

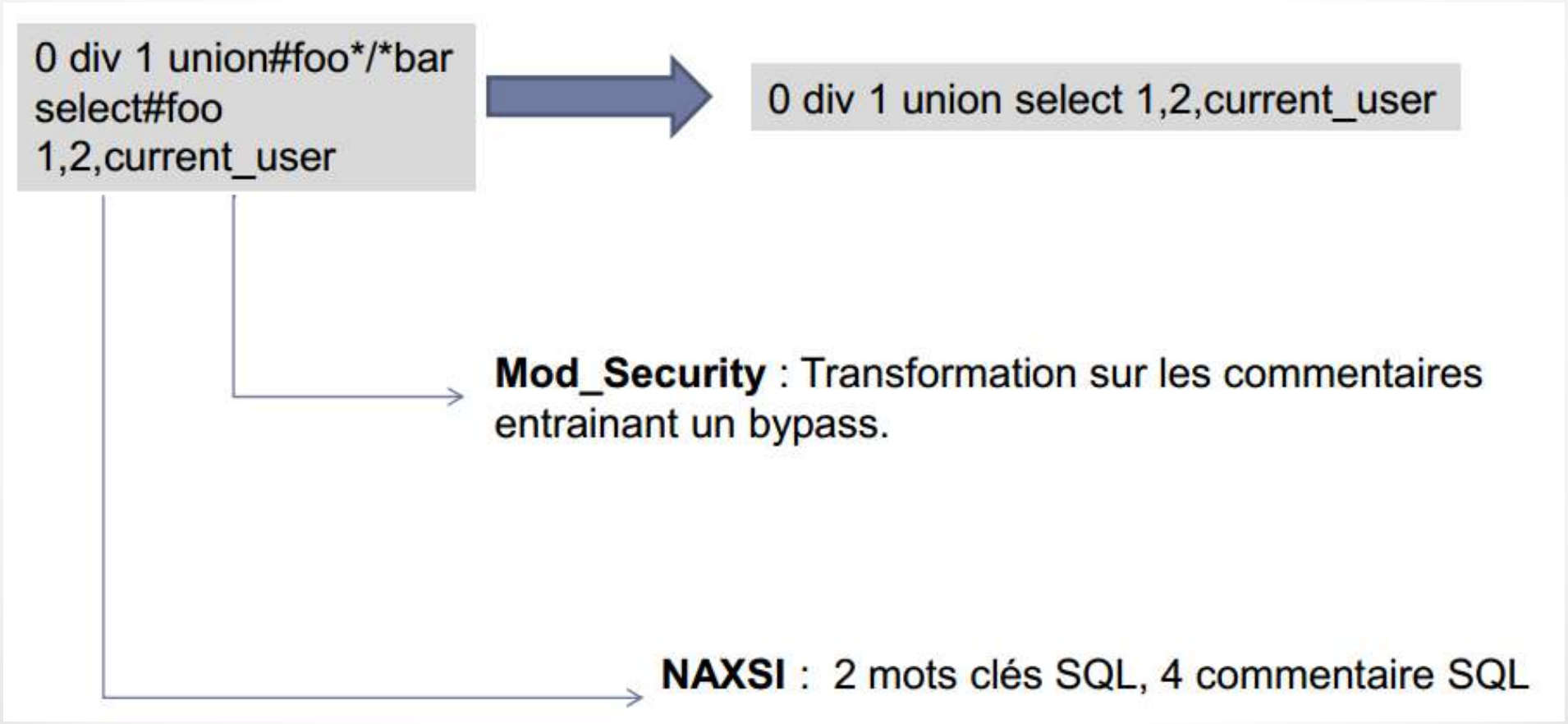
Learning Mode



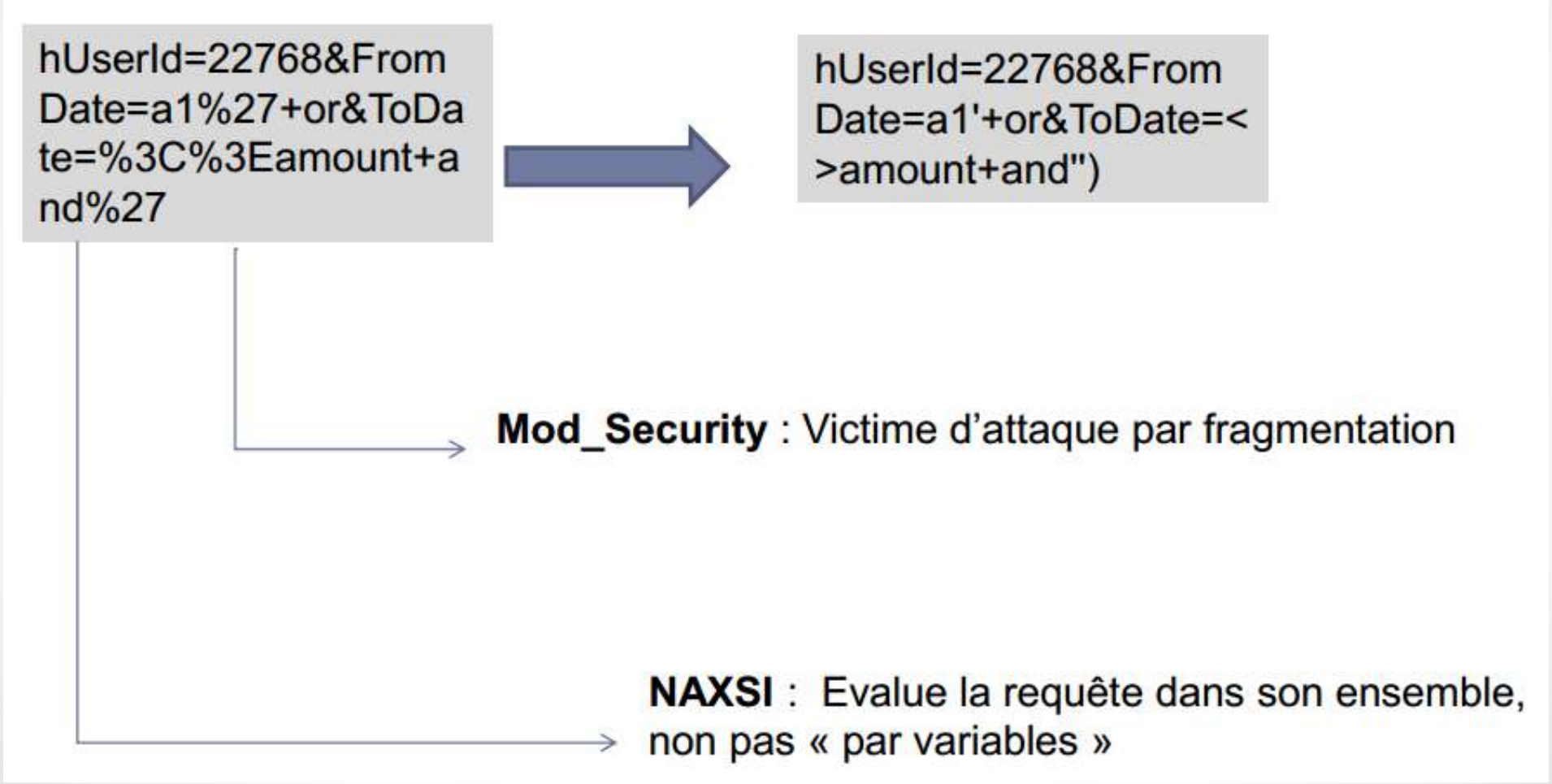
Learning Mode



Evasión WAF


0 div 1 union#foo*/*bar
select#foo
1,2,current_user
0 div 1 union select 1,2,current_user
Mod_Security : Transformation sur les commentaires entrainant un bypass.
NAXSI : 2 mots clés SQL, 4 commentaire SQL

Evasión WAF


hUserId=22768&From Date=a1%27+or&ToDate=%3C%3Eamount+and%27
hUserId=22768&From Date=a1'+or&ToDate=<>amount+and")
Mod_Security : Victime d'attaque par fragmentation
NAXSI : Evalue la requête dans son ensemble, non pas « par variables »

Creando Reglas

- Disable rule #1000 in GET argument named 'foo' :

  BasicRule wl:1000 "mz:$ARGS_VAR:foo";

- Disable rule #1000 in GET argument named 'foo' for url '/bar' :

  BasicRule wl:1000 "mz:$ARGS_VAR:foo|$URL:/bar";

- Disable rule #1000 in all GET arguments for url '/bar' :

  BasicRule wl:1000 "mz:$URL:/bar|ARGS";

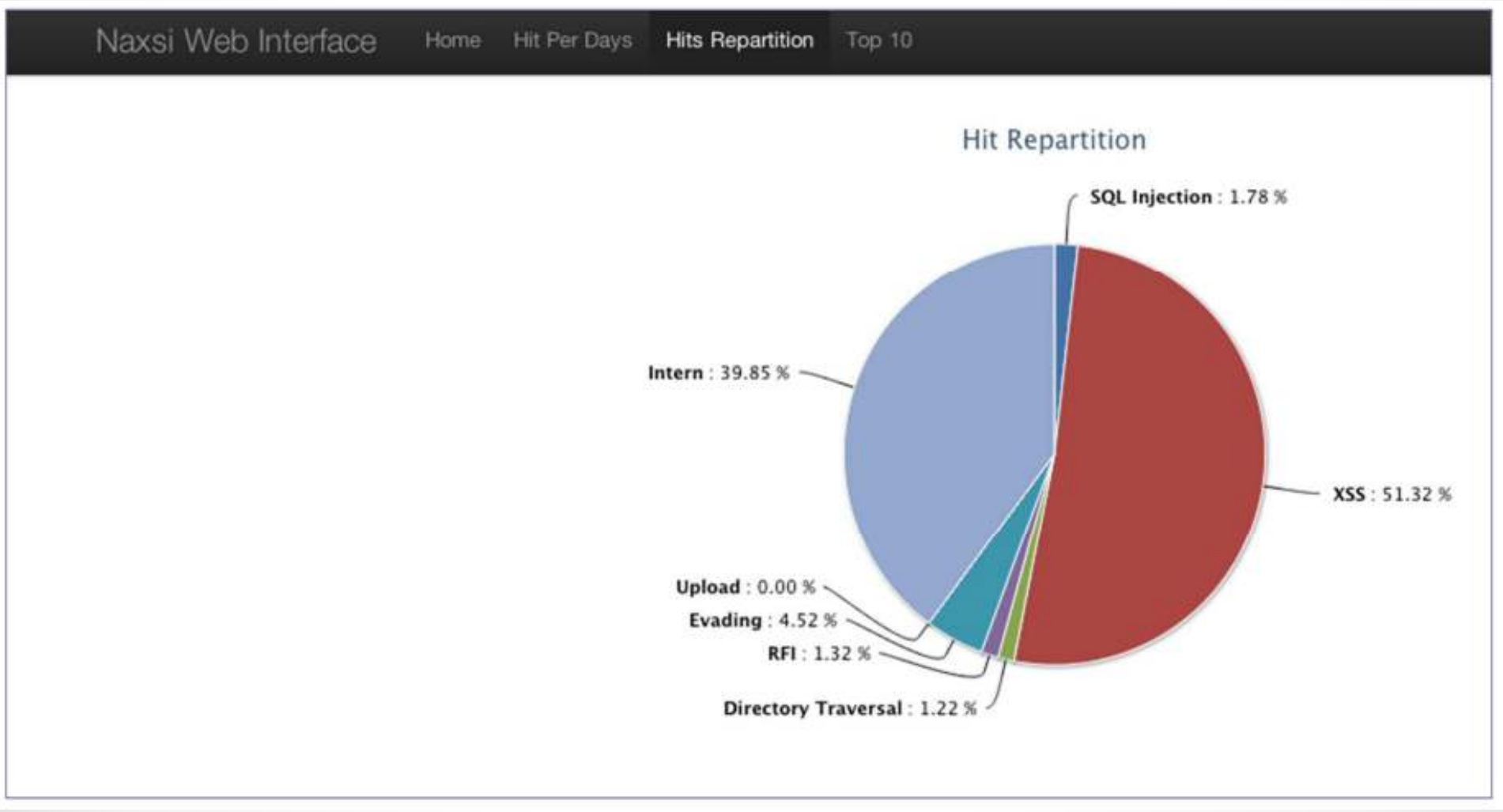
- Disable rule #1000 in all GET argument NAMES (only name, not content) :

  BasicRule wl:1000 "mz:ARGS|NAME";

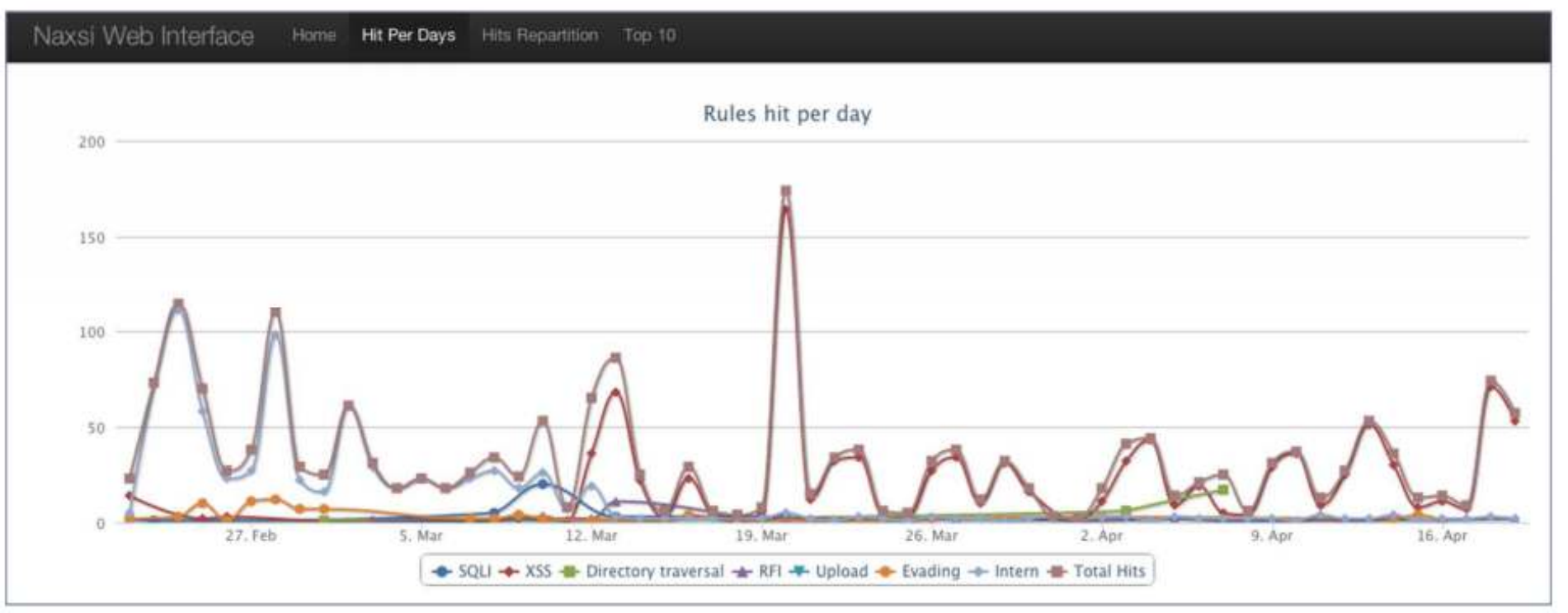
No tenemos tiempo para Reglas

- https://github.com/nbs-system/naxsi-rules

NAXSI - UI

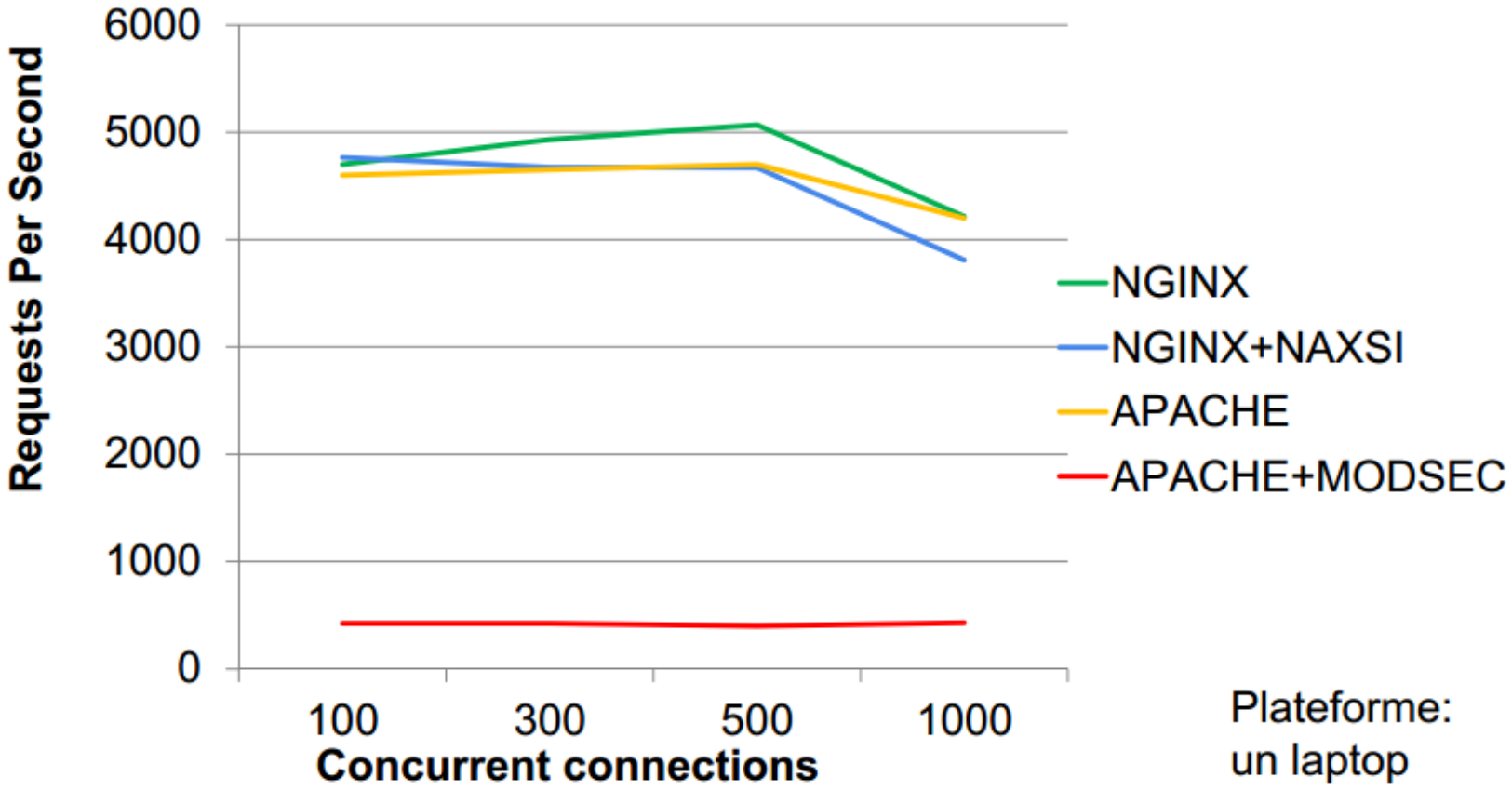

Naxsi Web Interface
Home
Hit Per Days
Hits Repartition
Top 10
Hit Repartition
SQL Injection : 1.78 %
Intern : 39.85 %
XSS : 51.32 %
Upload : 0.00 %
Evading : 4.52 %
RFI : 1.32 %
Directory Traversal : 1.22 %

NAXSI - UI


Naxsi Web Interface
Home
Hit Per Days
Hits Repartition
Top 10
Rules hit per day
200
150
100
50
0
27. Feb
5. Mar
12. Mar
19. Mar
26. Mar
2. Apr
9. Apr
16. Apr
SQLI
XSS
Directory traversal
RFI
Upload
Evading
Intern
Total Hits

Rendimiento

- Simple GET /index.html sans arguments



Rendimiento

	Nginx	Nginx+NAXSI	Diff (%)
Total time	1.151 s	1.271 s	9,4%
Req Per Sec	8687.21	7866.73	9,4%
Time Per Req (mean)	0.115	0.127	9,4%
Transfert Rate	1220.48	1198.45	1,8%

Retos de ModSecurity

- https://github.com/nbs-system/naxsi/wiki/naxsivsobfuscate

- Mas información:
- https://github.com/nbs-system/naxsi/