

Metacoretex-NG

Marc Llovet, Christian Martorella, Vicente Díaz

Edge Security
www.edge-security.com

Resumen. En este paper se presenta el *framework* Metacoretex-NG, *fork* de Metacoretex [1]. Se trata de un *software* creado para auditorías de bases de datos, que permite la fácil creación de *probes* por parte de los usuarios para ampliar las funcionalidades del sistema. Abandonado el proyecto original por su autor, decidimos retomararlo para revivir un *software* excelente y crear una referencia para la comunidad.

1 Introducción

Metacoretex es un *framework* para auditorías de bases de datos creado originalmente por visigoth de Security Centric Labs, cuya última versión disponible es la 0.8 de Septiembre de 2003. Este proyecto no se ha actualizado desde entonces. Tratándose de un excelente marco de trabajo, decidimos colaborar para aportar nuevas ideas, pruebas para las bases de datos, aportar funcionalidad y modernizar la interfaz de usuario. Sin embargo, el proyecto quedó parado debido a que lo abandonó su autor, así que decidimos crear un *fork* en el que implementar todas estas mejoras y ponerlo a disposición de la comunidad, para que el proyecto siga creciendo con el trabajo de todos y pueda llegar a ser una herramienta de referencia.

Metacoretex es un *framework*, lo que significa que es un marco de trabajo para análisis de vulnerabilidades de bases de datos. Permite de forma sencilla realizar una serie de pruebas contra distintos motores de bases de datos, desde los más sencillos como análisis de contraseñas y usuarios, hasta los más avanzados, como conseguir las SAM de un sistema Windows desde un MS SQL Server.

El usuario tipo de esta aplicación es variado: auditores, pen-testers, administradores de bases de datos, ... todo aquel que desee auditar la seguridad de una base de datos.

Nuestra intención es ampliar la funcionalidad de este proyecto. Queremos ampliar el número de bases de datos soportadas, el número de pruebas a realizar en todas ellas, modernizar la interfaz de usuario para facilitar la usabilidad y aumentar la eficiencia, optimizar las pruebas ya existentes para reducir su tiempo de ejecución y crear una comunidad que use esta herramienta y continúe mejorando el producto.

La implementación inicial del proyecto se hizo en Java. Esto proporciona un soporte multiplataforma y facilita la integración en la comunidad, ya que es un lenguaje ampliamente usado en todo tipo de proyectos libres. Nuestro *fork* ha seguido este camino, pero no descartamos un cambio de lenguaje en futuras versiones, pero manteniendo la importante característica de multiplataforma (python, .NET, ...).

2 Estado del arte

En este apartado analizamos algunas de las herramientas que existen en la actualidad y tienen funcionalidades similares a Metacoretex.

2.1 AppDetective [2]

Se trata de un análisis de vulnerabilidades de bases de datos en red. Es un producto de pago con numerosas pruebas para muchas bases de datos (MySQL, Oracle, MS SQL Server, DB2, ...). Es un producto interesante y completo, cuya mayor ventaja es la gran cantidad de sistemas que soporta y sus desventajas son ser un producto no abierto a la comunidad y no permitir añadir pruebas por parte de los usuarios. Creemos que un proyecto como Metacoretex podría ser una alternativa real: el soporte de una comunidad le proporcionaría tanto el soporte a gran cantidad de sistemas y pruebas como la creación de pruebas para las últimas vulnerabilidades tanto por la comunidad como por los propios usuarios.

2.2 NgSquirrel [3]

En este caso, se trata de toda una familia de productos para distintos motores de bases de datos, no de un único producto que los trate todos. Todos ellos son productos de pago. La filosofía de funcionamiento es parecida a todas las otras, sin soporte multiplataforma ni ejecución en red.

A pesar del tiempo que hace que apareció Metacoretex, vemos que en la actualidad no hay mucha más opciones, sin contar con aplicaciones libres disponibles para la comunidad. Es aquí donde el proyecto debe encontrar su lugar.

3 Metacoretex-NG y Trabajos Futuros

En este apartado explicamos los principales puntos a añadir en el *fork* de Metacoretex (bautizado como Metacoretex-NG) y las nuevas líneas de trabajo. Hay que destacar una vez más, el papel que toda la comunidad debe tener en este proceso, tanto probando el producto y dando sus impresiones, como contribuyendo activamente al desarrollo tanto del *framework* principal como de los probes de bases de datos.

3.1 Modernizar el framework

El proyecto original se diseñó hace dos años, con lo que la interfaz ha quedado muy desfasada. Aunque pueda parecer un punto menor, si un usuario no se siente cómodo y no es capaz de trabajar eficientemente con un *software*, no lo utilizará.

3.2 Ampliar el número de bases de datos soportadas

Ahora mismo soporta Oracle, MS Sql Server y MySQL, pero queremos soportar la mayoría de bases de datos con el apoyo de la comunidad (DB2, Postgres, Sybase, ...).

3.3 Ampliar número de probes y enfocarlas a pen-test

Tanto para bases de datos no soportadas como para las ya existentes. La idea es no sólo centrar la auditoría en consultas sobre la base de datos y acerca de la base de datos misma, sino acerca de cómo vulnerar la base de dato desde fuera y qué información se puede obtener al vulnerar la base de datos. Este punto es fundamental porque de este depende si la gente lo utiliza o no. Una de las razones por las cuales no tiene éxito el Metacoretex, es por que nadie lo actualiza.

3.4 Mejorar el sistema de informes

Este es un punto muy importante para todos los profesionales de la seguridad informática, y que puede ser el punto determinante que haga que se expanda su uso en entornos especializados.

La adopción de estos puntos, junto con la creación de una comunidad estable entorno al desarrollo de la aplicación, son los puntos clave para que Metacoretex-NG acabe siendo una de las Aplicaciones de referencia entre las de su tipo.

Referencias

1. <http://www.metacoretex.com>
2. <http://www.appsecinc.com/>
3. <http://www.nextgenss.com/squirreysql.htm>
4. <http://www.edge-security.com/metacoretex-ng/>