File Integrity Monitoring

The Good, Bad, Ugly, and the Future

OSSEC CON 2018

Dan Parriott – OSSEC Contributor

All About Me

Started using OSSEC in 2006

Most prolific poster to the mailing list

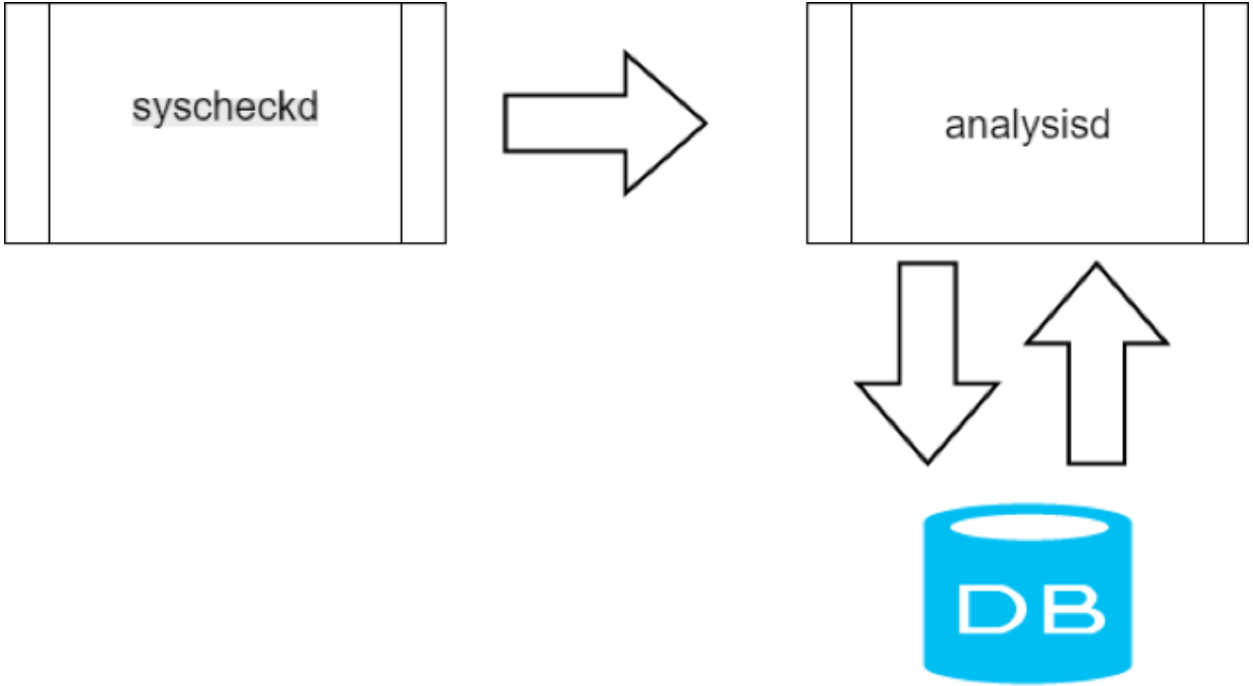
Occasional bug hunter

Sysadmin/security analyst

FIM Basics

“**File integrity monitoring (FIM)** is an internal control or process that performs the act of validating the integrity of operating system and application software files using a verification method between the current file state and a known, good baseline.“

Information Flow



syscheckd

Main process that does the thing.

Controls the syscheckd and rootcheck functionalities.

Syscheck

Computes file hashes

Checks file attributes

Syscheckd - The Good

Multiple hashes

Size, owner, permissions

Server compares new hashes to old

Realtime support on Windows and Linux

Registry support on Windows

Diffs of changed files

Syscheckd - The Bad

The hashes are md5 and sha1

Does file deletion detection work?

Reports of registry checking not working properly on 64bit Windows

Full scan blocks realtime

Syscheckd - The Ugly

The database is a flat file, nothing is deleted

Rootcheck will block syscheck

Strange timing can cause false positives

Reports of syscheck missing changes

Syscheckd - The Future

SHA256 from libsodium

Blake2b?

Architectural shift?

Linux extended attributes

Rootcheck

Searches files for suspicious signatures

Can check config files for misconfigurations

Rootcheck - The Good

CIS checks for Debian/Redhat

Trojan/rootkit checks

Windows registry monitoring

Rootcheck - The Bad

Does it work?

Do the Windows registry checks work on win64?

Configuration is unique

Checks are out of date

Rootcheck - The Ugly

No state keeping in the checks

Does not understand containers

Rootcheck - The Future

It will be tested

64bit Windows builds?

Lua everywhere

New configuration syntax?

Analysisd

Receives information from syscheckd

Compares information to a baseline database

Tracks changes

Sends alerts

md5 whitelist

Analysisd - Possible Future Features

Extend md5 whitelist

Sqlite3 database backend

Allow hashes and file names to be used in active response

Access external services for a second opinion

Gratuitous Tech Shot

Learn More About OSSEC

- OSSEC GitHub Site
- OSSEC Download Page
- The OSSEC community on Slack
- Subscribe to monthly OSSEC newsletter
- Follow the OSSEC Project on Twitter