



to Pluginfinity and Beyond 🚀

✓ **feat(falco): maintainers track**

[Browse files](#)

Signed-off-by: Jason Dellaluce <jasondellaluce@gmail.com>

Signed-off-by: Leonardo Grasso <me@leonardograsso.com>

🔗 **kubecon-cnc-eu** (#2022)



2 people authored and **poiana** committed



About us



Jason Dellaluce
Open Source Engineer
Falco Maintainer

jasondellaluce 

jasondellaluce 



Leonardo Grasso
Open Source Engineer
Falco Maintainer

 leogr

 leogrease



to



and Beyond

Jason Dellaluce & Leonardo Grasso



In this talk...

The Falco Maintainers Track



What's Falco

A brief
introduction

Pluginfinity

The
game-changer!

Plugin registry

Getting involved

What's new?

Last year updates

Plugin SDKs

What's your
favorite lang?

What's next?



to



and Beyond

Jason Dellaluce & Leonardo Grasso



What is Falco?



A security camera detecting unexpected behavior, intrusions, and data theft in real time



The cloud-native runtime security project

The de facto Kubernetes threat detection engine



to

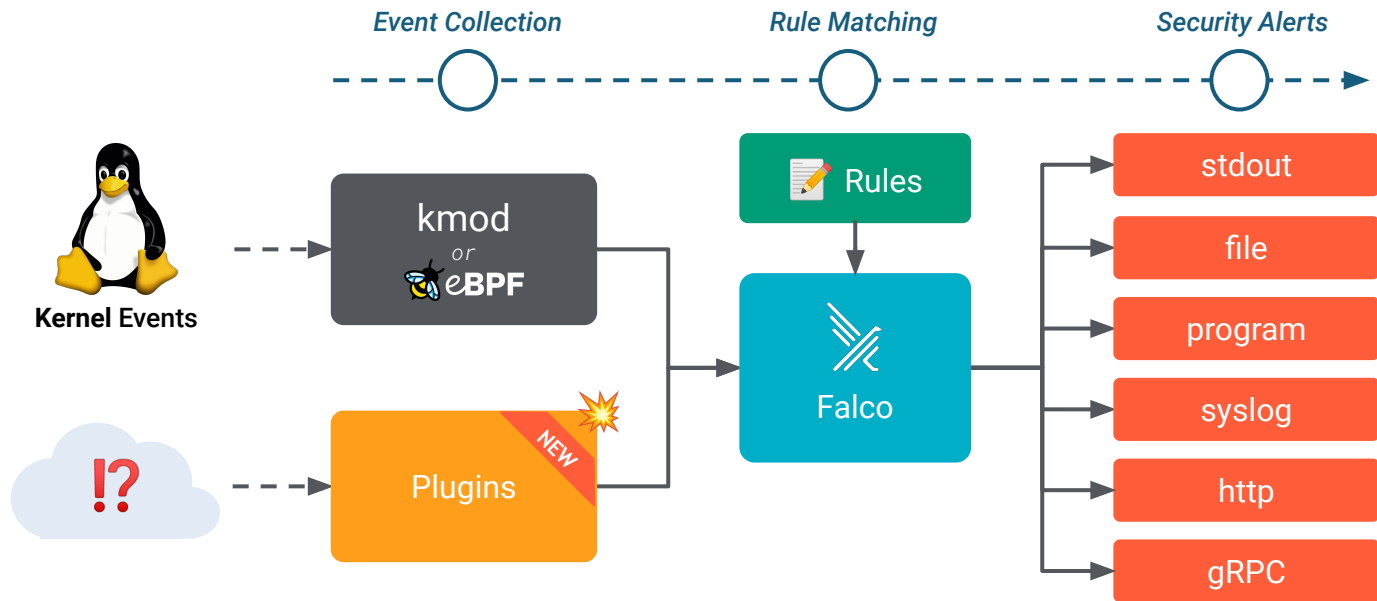


and Beyond

Jason Dellaluce & Leonardo Grasso



How does Falco work? 🤖



to



and Beyond

Jason Dellaluce & Leonardo Grasso



Falco's Journey



Falco haunted
its first prey

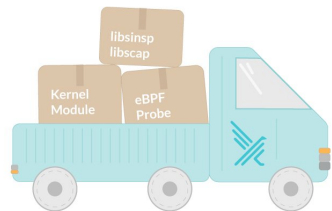
2016

First ever Falco
Community Call

2018

Promoted to CNCF
incubating project

2020



to



and Beyond

Jason Dellaluce & Leonardo Grasso



What 's new?

2021 – 2022



Project updates!



New Release Cadence



Starting from Falco 0.30 (Sept 2021)



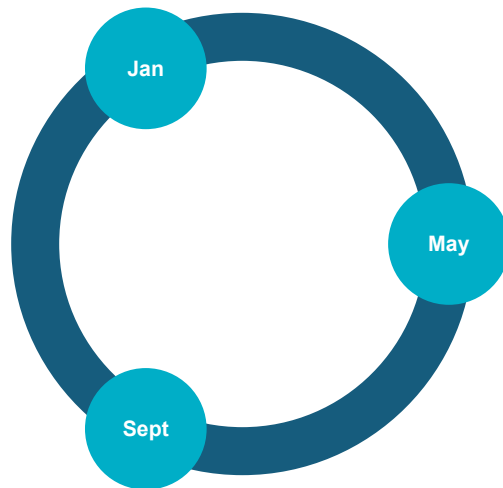
Falco now gets released
3 times per year



The current schedule sees
a new release by the end of
January, May, and September



Hotfix releases can happen
whenever needed



to



and Beyond

Jason Dellaluce & Leonardo Grasso



New Security Rules



File System

- Create Hardlink Over Sensitive Files
- Create Symlink Over Sensitive File
- Create files below dev
- Debugfs Launched in Privileged Container

Privilege Escalation

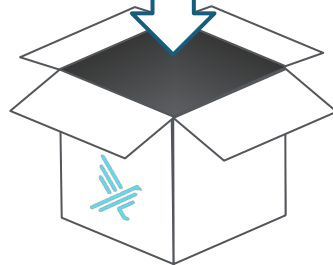
- Non sudo setuid
- Linux Kernel Module Injection Detected
- Polkit Local Privilege Escalation Vulnerability
- Sudo Potential Privilege Escalation
- Unprivileged Delegation of
- Page Faults Handling to a Userspace Process

Crypto Miners

- Detect crypto miners using the Stratum protocol
- Detect outbound connections to common miner pool ports

Containers

- Change thread namespace and Set Setuid or Setgid bit
- Container Drift Detected
- Launch Package Management Process in Container
- Launch Remote File Copy Tools in Container



to



and Beyond

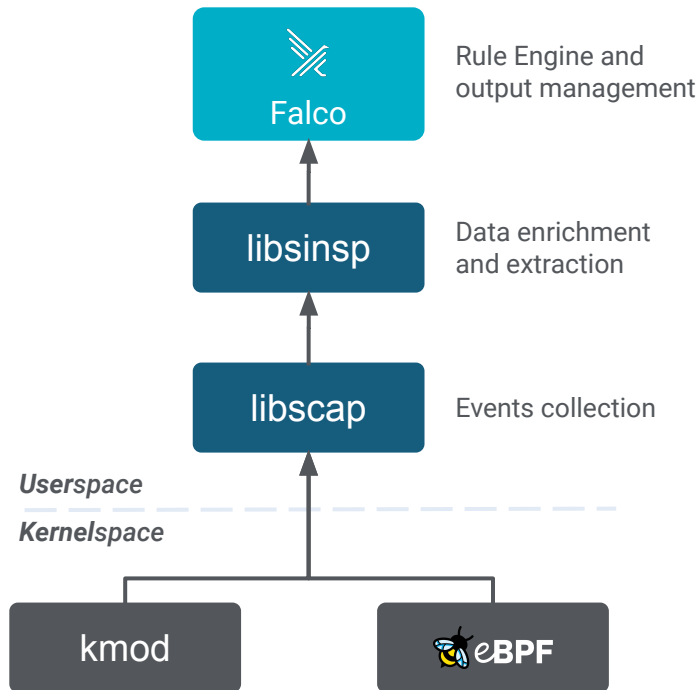
Jason Dellaluce & Leonardo Grasso



Libs and Drivers: Renovations



- Contribution of the drivers and the libs
to Falcosecurity in Feb 2021
- The great refactor TM
 - Clean up legacy code & dependencies
 - Performance optimizations
- Increased support and stability
 - eBPF: wider kernel compatibility
 - ARM64: stable support is coming



to



and Beyond

Jason Dellaluce & Leonardo Grasso



Libs and Drivers: Security



- [CVE-2021-33505](#), [CVE-2022-26316](#)

follow our [Security Advisories](#)!

- New security-critical syscalls

`userfaultfd`

`openat2`

`open_by_handle_at`

`copy_file_range`

`io_uring_setup`

`Io_uring_enter`

`io_uring_register`

`execveat`

`clone3`

`capset`

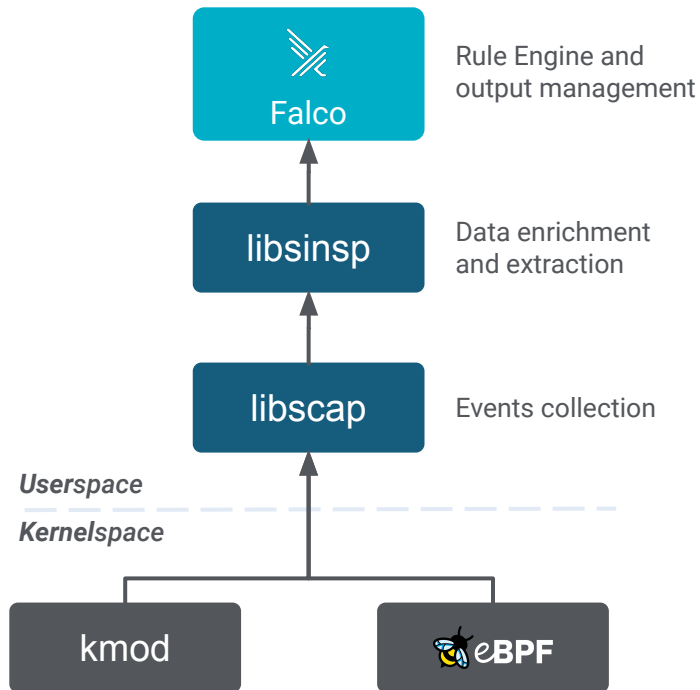
`mprotect`

`mlock`

`munlock`

`mlockall`

`munlockall`



to



and Beyond

Jason Dellaluce & Leonardo Grasso

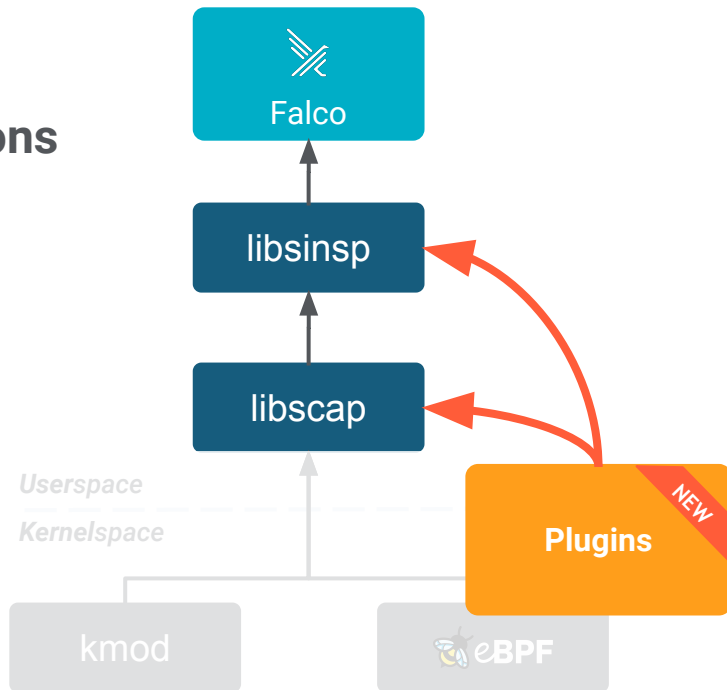


The New Plugin System

- 💡 Plugins extend the capabilities of Falco
- 100 🚀 Standard way for new features and integrations
- 💣 Dramatically enlarge Falco's use cases
- ☁️ Bring Runtime Security to new domains

↓

**Better option for Threat
Detection in the Cloud**



to



and Beyond

Jason Dellaluce & Leonardo Grasso

Pluginfinity



(and beyond)



What are plugins?

Plugins are...



Dynamic shared libraries

.so in Unix

.dll in Windows



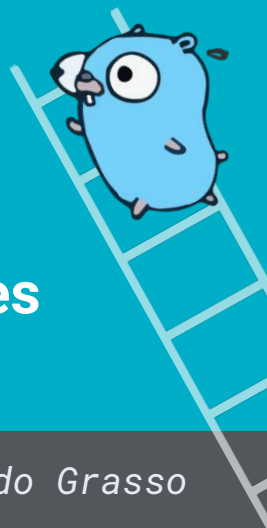
**Compliant to a simple API
of ~20 C-symbols**



**Developed in
any language**



**Modular extensions
of Falco's capabilities**



to

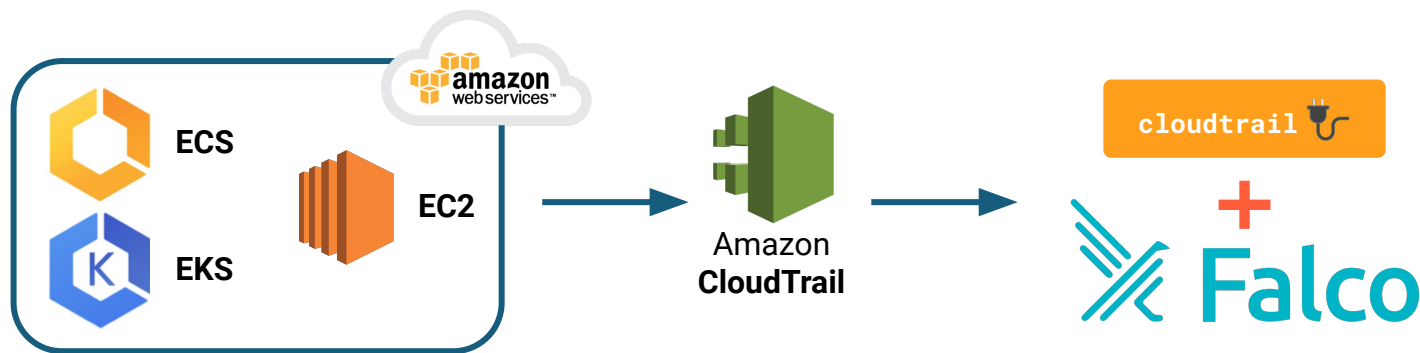


and Beyond

Jason Dellaluce & Leonardo Grasso

Plugins: *event sourcing* capability

- Add new data sources to Falco (local or remote)
- Provide new streams of events to the Falco Rule Engine
- Dramatically expand the use cases of Falco in Cloud Security



to



and Beyond

Jason Dellaluce & Leonardo Grasso



Plugins: *field* extraction capability

- Extract more information from events
- The new data fields can be used to write new Falco rules
 - As filtering condition
 - As rule output
- Can be generic or tied to specific data sources

```
rule: Console Login Without MFA
desc: Detect a console login without MFA
condition:
  ct.name="ConsoleLogin" and not ct.error exists
  and ct.user.identitytype!="AssumedRole"
  and json.value[/responseElements/ConsoleLogin]="Success"
  and json.value[/additionalEventData/MFAUsed]="No"
output:
  Detected a console login without MFA
  (requesting user=%ct.user,
   requesting IP=%ct.srcip,
   AWS region=%ct.region)
priority: CRITICAL
source: aws_cloudtrail
```

(from the official **default ruleset** of the **CloudTrail** plugin)



to



and Beyond




Jason Dellaluce & Leonardo Grasso

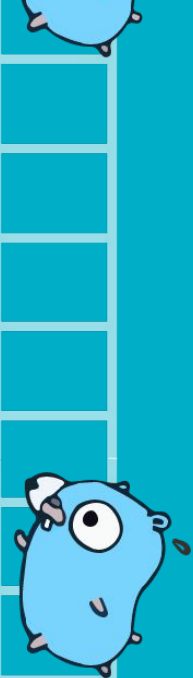


Plugin SDKs

How can I get started?

Community-supported
language SDKs 

- SDK Go – Available! 
- SDK C++ – Still a WIP 
- More will come... 



to



and Beyond

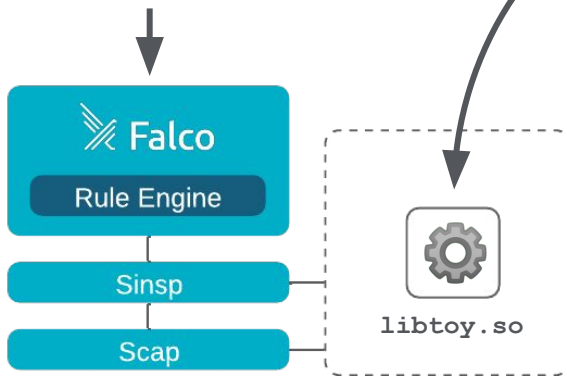
Jason Dellaluce & Leonardo Grasso



Plugin SDK Go: Keep it Simple™

falco.yaml

```
plugins:
- name: toyplugin
  library_path: libtoy.so
  init_config:
    name: Buzz Lightyear
  load_plugins: [toyplugin]
```



toyplugin.go

```
type ToyPlugin struct {
    plugins.BasePlugin
    Cfg struct {
        Name string `json:"name"`
    }
}

func (t *ToyPlugin) Init(c string) error {
    return json.Unmarshal([]byte(c), &t.Cfg)
}

func (t *ToyPlugin) Destroy() {
    println("👋 See you ", t.Cfg.Name)
}
```

👉 github.com/falcosecurity/plugin-sdk-go 👉



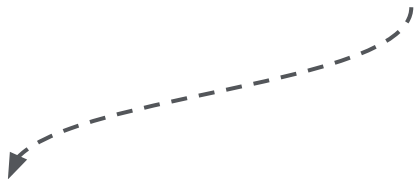


github.com/falcosecurity/plugin-sdk-go



pkg/sdk/plugins/**source**

```
type Plugin interface {  
    Info() *plugins.Info  
    Init(config string) error  
    Open(params string) (Instance, error)  
}
```

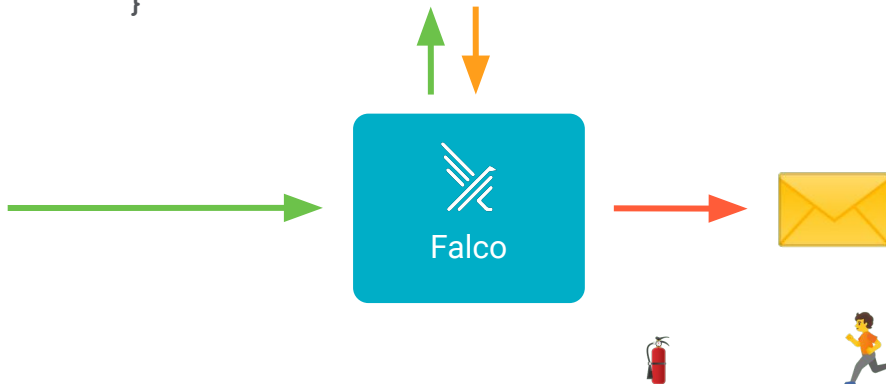


```
type Instance interface {  
    NextBatch(p sdk.PluginState,  
             e sdk.EventWriters)  
             (int, error)  
}
```



pkg/sdk/plugins/**extractor**

```
type Plugin interface {  
    Info() *plugins.Info  
    Init(config string) error  
    Fields() []sdk.FieldEntry  
    Extract(r sdk.ExtractRequest,  
           e sdk.EventReader) error  
}
```



to



and Beyond

Jason Dellaluce & Leonardo Grasso



Plugin Registry

Records of plugins officially
acknowledged by the community

👉 github.com/falcosecurity/plugins 👉

Goals



Respecting technical constraints



Unique Plugin ID assignment



Coordination on **source names**



**Help developers to share and
promote their plugins in the
community**



to



and Beyond

Jason Dellaluce & Leonardo Grasso



Lots of plugins in the community



The ecosystem is growing fast since february... 🤖



Kubernetes
Audit Logs



Amazon
CloudTrail



Docker



Seccomp
Agent



System Log

...and you can be next!

We are looking forward to see new integrations and contributions



to



and Beyond

Jason Dellaluce & Leonardo Grasso

What 's next?



Sneak peek at future developments



Rule & plugin distribution



The plan 🐱

- ◆ A tool for **downloading** and **installing** plugins and rules
- ◆ **Continuous** ruleset updates
- ◆ Resurrecting `falcoctl` 💀



to



and Beyond

Jason Dellaluce & Leonardo Grasso



eBPF

The next generation probe
is coming...

Highlights ✨

- ◆ **CO-RE** support
- ◆ BPF **ring buffer**
- ◆ Less **BPF helpers**
- ◆ Native **multi arch support**

Development has started!

See the official **proposal** ➡



to



and Beyond

Jason Dellaluce & Leonardo Grasso



Join the community 🤗

#falco channel on the



Kubernetes Slack 🧑💻

Falco **community call**

every Wednesday 🤝

✉ Mailing list ✉

cncf-falco-dev@lists.cncf.io

👉 github.com/falcosecurity/community 👉



to



and Beyond

Jason Dellaluce & Leonardo Grasso



Falco
to Pluginfinity and Beyond
Jason Dellaluce & Leonardo Grasso

Thank you!



No Emojis Were Harmed