The HOE Stack: OpenNMS + Helm + Elasticsearch

Jeff Gehlbach
OpenNMS User Conference Europe
21 Sep 2018 · Munich

Jeff Gehlbach

Director of Applied Innovation at The OpenNMS Group

Spelling SNMP since 2000.
Contributing to OpenNMS since 2005.
Working for The OpenNMS Group since 2007.

openNMS®

Agenda

1. **The way things were**
   A bit of background regarding event-heavy use cases and OpenNMS
2. **The way things are**
   Why we can't just go back
3. **The challenge**
   What is the problem that needs solving, anyway?
4. **A real-world example**
   Overview of a real customer environment
5. **How to eat a tonne of grain**
   Decomposing the problem
6. **Integrating the pieces**
   Making it all work together
7. **Pitfalls**
   Lessons learned from actually fielding the solution
8. **Version notes**
   Details vary
9. **Conclusion**
   Wherein I conclude
10. **Q&A**
    Wherein you say things and I say other things

The way things were

Once upon a simpler time, in Bavaria...

Fork-hoe depiction in Der Rebmann (the vine-dresser). Jost Amman, Das Ständebuch, 1568
Public Domain, https://commons.wikimedia.org/w/index.php?curid=249439

OpenNMS 1.2 / circa 2006

E

Events

Originally had the concept of acknowledgment

Results 1-10 of 38988
1 2 3 4 5 Next Last

Ack	ID	Severity	Time	Node	Interface	Service	Ackd
☐	39480	Normal [+] [-]	4/23/05 11:30:23 AM [<] [>]	mrmakey.opennms.com [+] [-]	172.20.1.1 [+] [-]		
			Linksys Event: @out TCP from 172.20.1.10:47964 to mail.befunk.com(66.139.77.96):21.				
☐	39479	Normal [+] [-]	4/23/05 11:29:26 AM [<] [>]	mrmakey.opennms.com [+] [-]	172.20.1.1 [+] [-]		
			Linksys Event: @out UDP from 172.20.1.11:123 to clock1.unc.edu(152.2.21.1):123.				
☐	39478	Normal [+] [-]	4/23/05 11:29:20 AM [<] [>]	mrmakey.opennms.com [+] [-]	172.20.1.1 [+] [-]		
			Linksys Event: @out TCP from 172.20.1.10:47960 to mail.befunk.com(66.139.77.96):443				

A

Alarms

Lifecycle entities — “events tha tmatter”. Basic de-duplication via reduction key.

Results 1-10 of 16
1 2 Next Last

Ack	ID	Severity	Node	Interface	Service	Count	Last Event Time	First Event Time
			Ackd	Ackd Time				
☐	2	Normal [+] [-]	mrmakey.opennms.com [+] [-]	172.20.1.1 [+] [-]		38714	4/23/05 11:17:16 AM [<] [>]	4/19/05 4:45:10 PM [<] [>]
			Linksys Event: @out TCP from 172.20.1.204:65247 to 198.128.246.160(198.128.246.160):80.					
☐	9	Major [+] [-]	172.20.1.201 [+] [-]	172.20.1.201 [+] [-]	SSH [+] [-]	23	4/23/05 7:44:32 AM [<] [>]	4/21/05 3:57:30 PM [<] [>]
			SSH outage identified on interface 172.20.1.201.					
☐	10	Cleared [+] [-]	172.20.1.201 [+] [-]	172.20.1.201 [+] [-]	SSH [+] [-]	23	4/23/05 7:45:08 AM [<] [>]	4/21/05 3:59:52 PM [<] [>]
			The SSH outage on interface 172.20.1.201 has been cleared. Service is restored.					
☐	7	Critical [+] [-]	172.20.1.201 [+] [-]	0.0.0.0 [+] [-]		3	4/21/05 6:05:41 PM [<] [>]	4/20/05 6:36:26 PM [<] [>]
			Node 172.20.1.201 is down.					
☐	8	Cleared [+] [-]	172.20.1.201 [+] [-]	0.0.0.0 [+] [-]		3	4/22/05 9:51:38 AM [<] [>]	4/21/05 9:21:23 AM [<] [>]
			Node 172.20.1.201 is up.					

Limitations of original events / alarms implementation

- External events had to be transported to central OpenNMS listener
- Correlation functionality was limited to a single reduction-key
- Events and alarms were persisted only into relational database (PostgreSQL)
- Automations were limited to SQL-based triggers and actions
- Lack of a supported API for external access to events and alarms
- Event / alarm browsers in OpenNMS web UI frankly pretty bad

The way things are

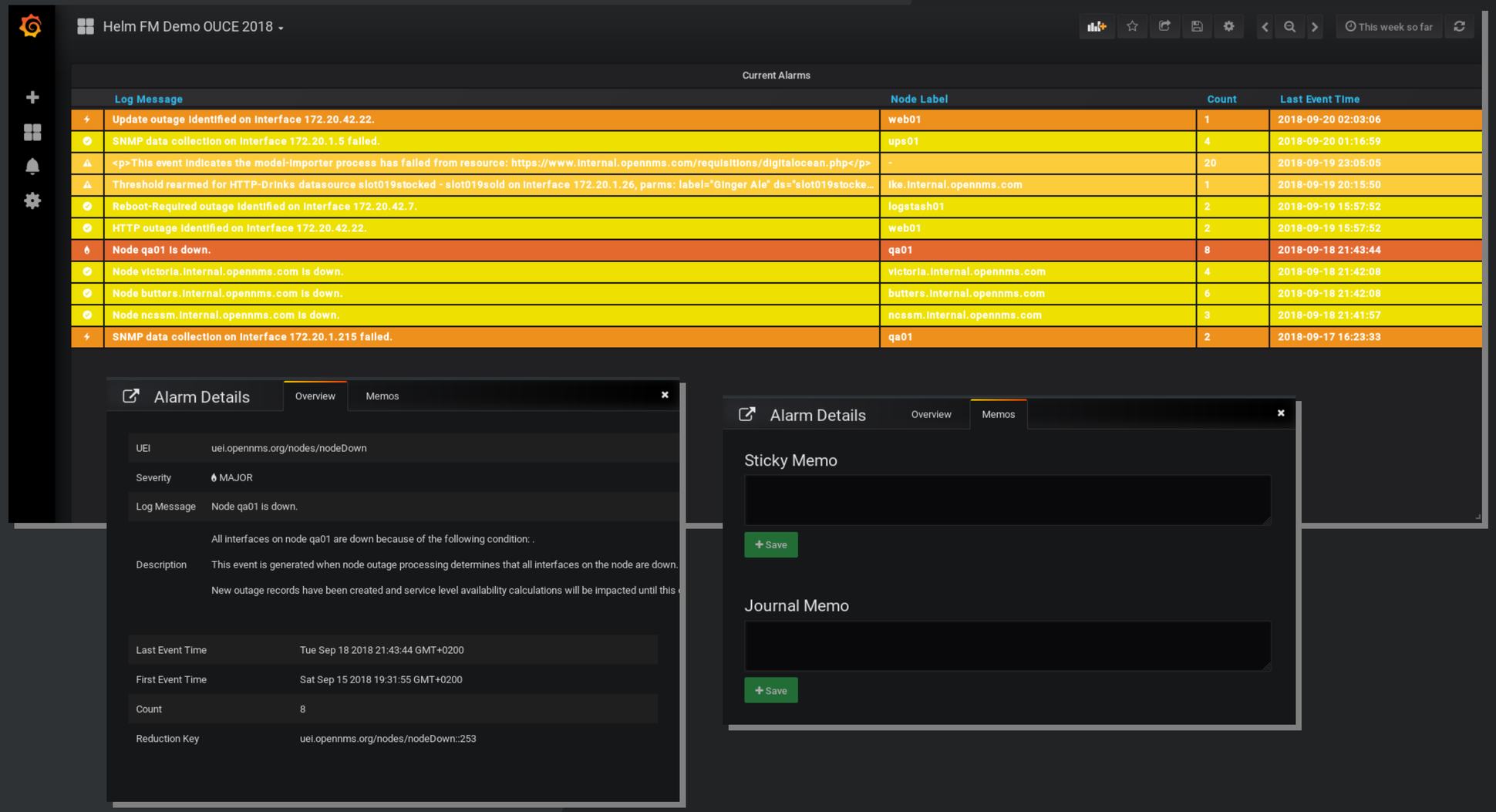
Every time is simpler compared to some other time

Improvements to events / alarms implementation

- External events may now arrive via Minion, mitigating complex network topology
- Correlation possibilities opened up via Drools integration
- Events and alarms may be streamed to Elasticsearch for archival
- First-rate REST API provides external access to events and alarms
- Faults data source for Helm / Grafana provides an improved alarm browser

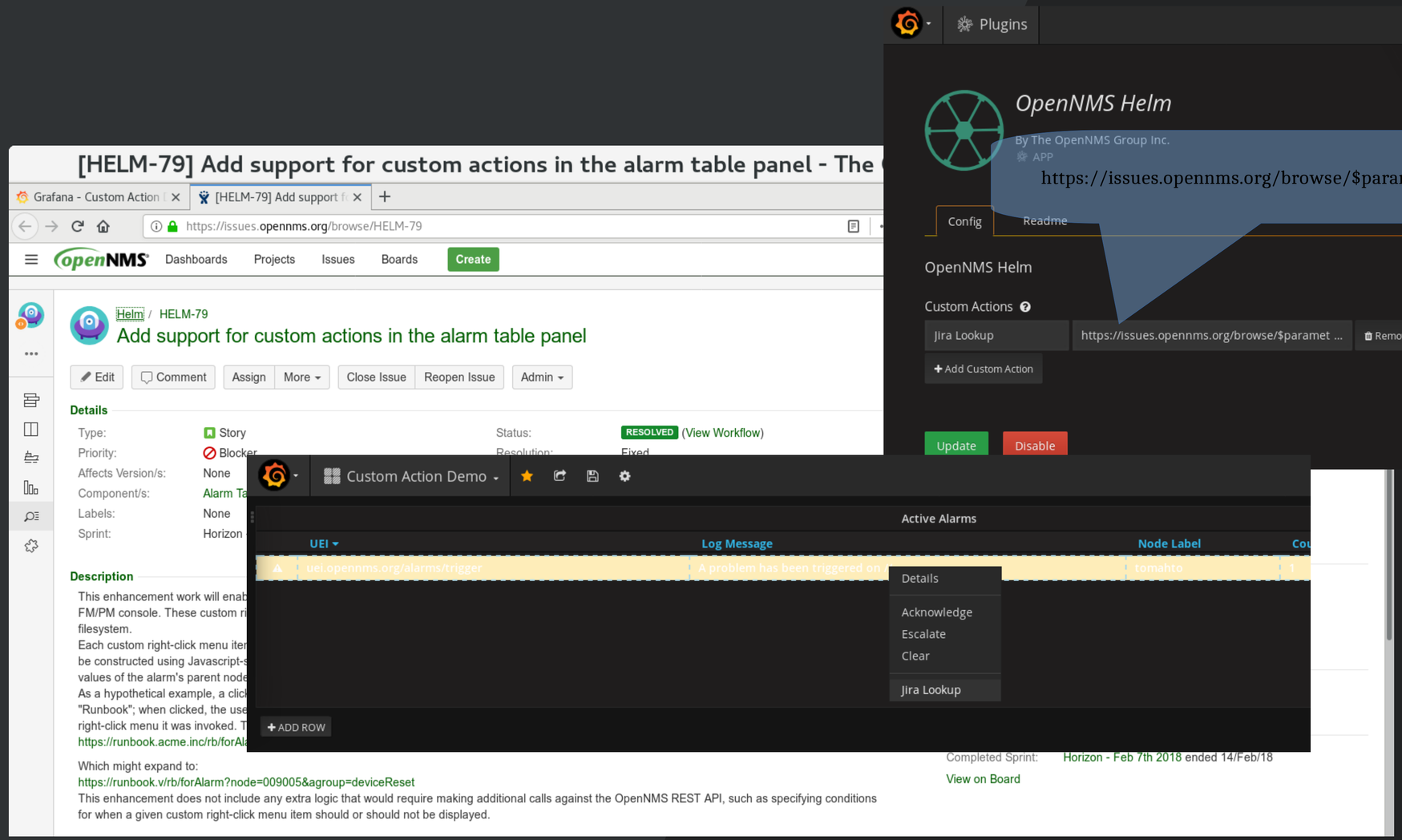
OpenNMS Helm Faults Data Source for Grafana

- Uses alarms REST API
- Provides Alarm Table panel for Grafana
- Supports alarm actions
  - Acknowledge
  - Unacknowledge
  - Escalate / Clear
  - Edit alarm memos
  - Custom actions



OpenNMS Helm Alarm Table Custom Actions

- Specify any URL
- Substitute in alarm parameters to create a click-across integration with your favorite third-party web thing
- Action appears in right-click menu of any alarm having all the required parameters



The challenge

It’s not getting simpler any time soon

NASA - http://earthobservatory.nasa.gov, Public Domain, https://commons.wikimedia.org/w/index.php?curid=768621

Data volumes are growing

- Event volume in a typical event-focused customer environment is increasing
- The operational OpenNMS RDBMS makes a poor data warehouse
  - Systems such as Netcool use an in-memory DB along with “gateways”
  - Our architecture does not afford us this luxury
  - Keeping too many events creates a drag on OpenNMS performance
- Customers want analysis and auditing of historical events
- How do we handle this need without sacrificing performance?

A real-world example

Overview of a real customer environment

Case study overview — back end

- Customer is a popular provider of in-flight WiFi Internet access
- 100% SNMP trap-driven management workflow
- Every flight is book-ended with “hello” / “goodbye” traps
- In between, we get periodic “heartbeat” traps and other kinds of traps
- Legacy event management platform is Netcool
  - Single mttrapd probe in terrestrial collocation facility
  - Single ObjectServer in same facility
  - Netcool Impact rules approximate a state machine using these traps
  - A “dark flight” is bad news for revenue. Priority #1 is to recognize these.
- All software that flies is subject to extremely strict change controls from civil aviation authorities (chiefly FAA)

Case study overview — front end

- Shift operators require a familiar event management UI like Netcool’s AEL
- Shift managers need ability to audit history of each alarm
- Internal customers must be able to report on months of historical events

Case study overview — technical constraints

- Customer faced an enterprise-wide mandate to migrate IT to AWS
  - Also a rigid mandate on choice of DevOps pipeline
  - The chosen pipeline was a poor fit for OpenNMS
    - And virtually impossible to duplicate in a lab
- Airborne systems send their traps to a hard-coded IPv4 address
  - Changing this would take months and cost multiple 100K USD
  - Hits mttrapd ProbeServer via a physical load balancer
- No ground-to-air IP traffic is allowed

How to eat a ton of grain

Decomposing the problem

Decomposing the problem: deployment

- Minion on VMware VM; Balance via pipeline → EC2, RDS, SQS
  - Mind the configuration file mutability

Decomposing the problem: trap intake and correlation

- Eventconf XML gets us a long way down the road
- Drools and a third-party incumbent platform close the gap

Decomposing the problem: event archival to Elasticsearch

- Customer using Elastic.co AWS-hosted cluster
- Aggressive event cleanup in DB (TTL measured in hours)

Decomposing the problem: alarm visualization for operators

- Grafana Helm Faults data source plugin

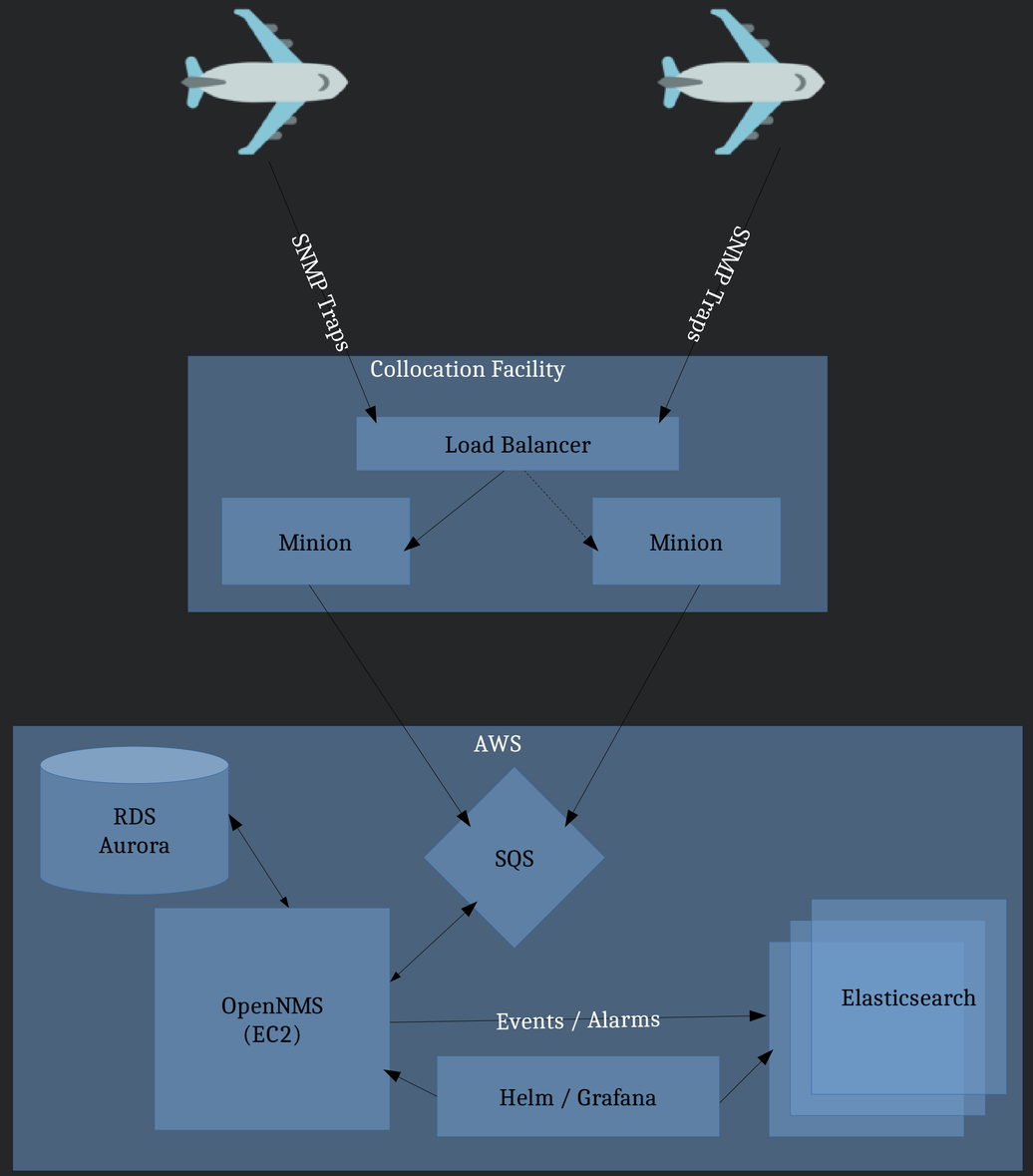
Decomposing the problem: alarm reporting & analysis

- Grafana Elasticsearch data source plugin
- Kibana

Integrating the pieces

Making it all work together

Case study overview — diagram



Stack component version requirements

Helm

- Helm 2.0
- Grafana 5.x

OpenNMS

- Horizon ≥ 21
- Meridian ≥ 2017

Elasticsearch

- ≥ 5.x

Pitfalls

Lessons learned from actually fielding the solution

Pitfalls

- Alarm change notifier plugin problematic with high event volumes
- Uninstalling this plugin is tricky once it has become a problem
- Back-pressure from Elasticsearch on Eventd via Alarmd can cause bottlenecks
  - Alarmd is now multithreaded as a result of experience from the case study
  - Under-sizing your Elasticsearch cluster worsens the problem
- Under-sizing the RDBMS (PostgreSQL or RDS Aurora) is very bad for performance

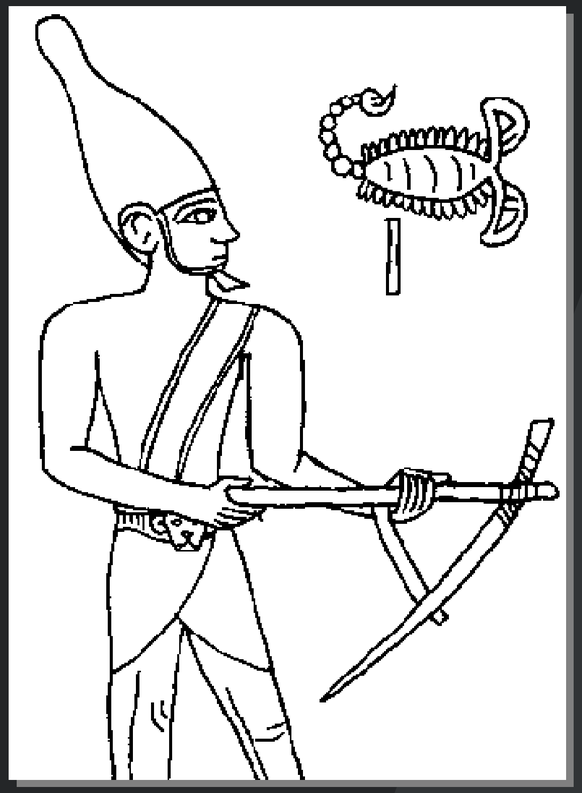
Version notes

Details vary

Version notes

- Horizon 22 reworks Elasticsearch event / alarm forwarding. See admin guide.
- For events and alarms alone, Elasticsearch 5 is fine.
- For flows, Elasticsearch 6 is required, so go straight for that if you can.
- Helm 2.0 requires Grafana 5.0.

Conclusion



Q&A