



PT Container Security

Product roadmap

Container security from Positive Technologies

Containers and cybersecurity everywhere



69%

- Level of Kubernetes integration among organizations

31%

developers use Kubernetes globally

• 25%

of teams don't scan their images for vulnerabilities

74%

Share of organizations using DevSecOps

94%

specialists have experienced at least one security incident related to containers or Kubernetes

50%

- of DevOps specialists postpone the deployment
- of applications because of
- container or Kubernetes security issues

Practical cybersecurity solutions



21 years

of security research
and development

1,500+

security engineers,
developers, analysts,
and other specialists

250+

experts at our security research center

200+

zero-day vulnerabilities discovered every year

250+

security audits
of corporate systems
performed annually

50%

of all industrial and
telecom vulnerabilities have been
discovered by our experts

▪ Cybersecurity products and solutions ▪ Security audits

▪ Incident investigation

▪ Threat research

But things often go wrong



27%

No Kubernetes cluster
configuration control policies

Excessive container
privileges

25%

Unsanctioned launch of third-
party images

24%

Violation of the confidentiality of
secrets

Compromised container
image

20%

Violations of the access
model
to the Kubernetes cluster

Access to the quality
control console

23%

Exploitation of vulnerability in
the container

19%

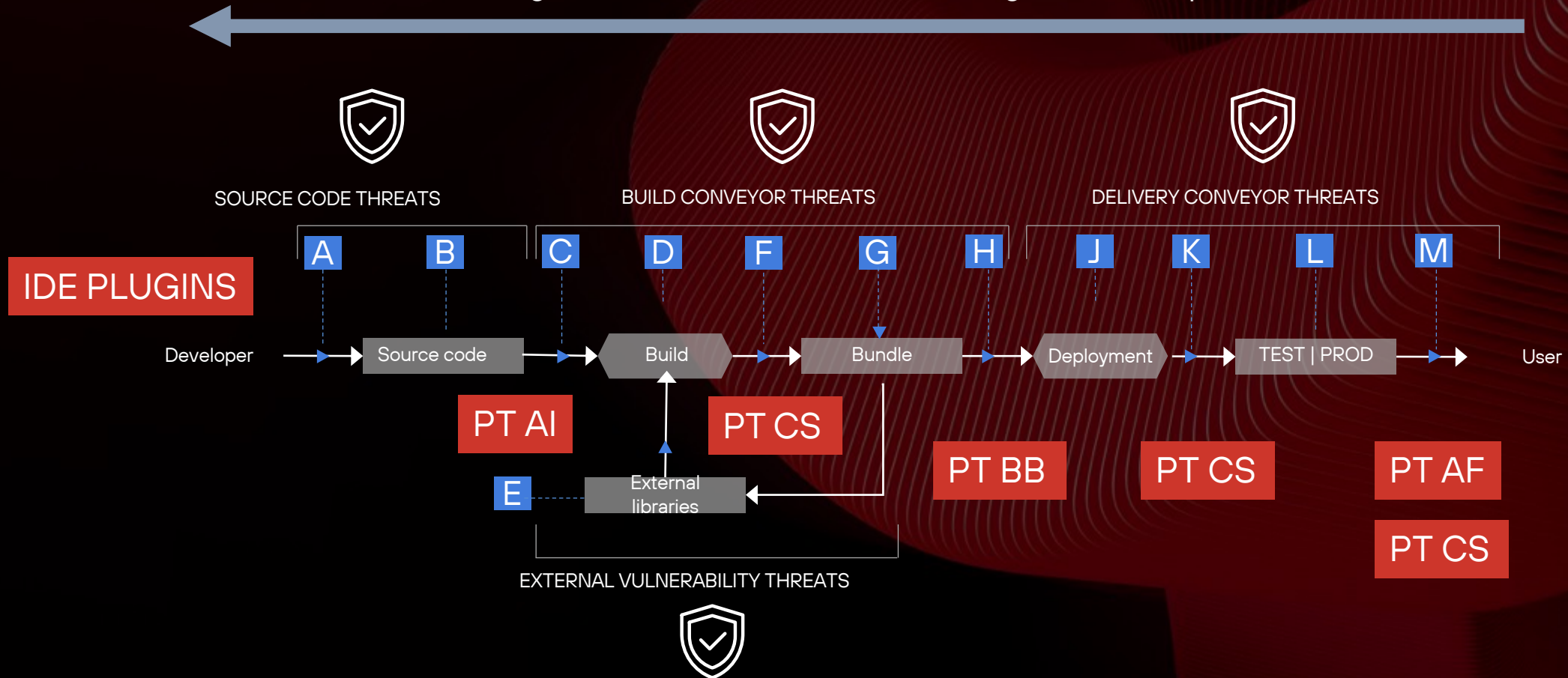
Violation of container network
policies

14%

Violation of access
to resources in the cluster

Secure development in containers

Shifting the focus of checks to earlier stages of development



Why PT Container Security?

PT Container Security key features



User-friendly, predictable, and manageable right out of the box.

- Single search window and settings suggestions.
- Ultimate scalability: system functions can be scaled separately (native to K8s and user-friendly).
- Smart analysis based on metrics shows you what's happening and what to do.

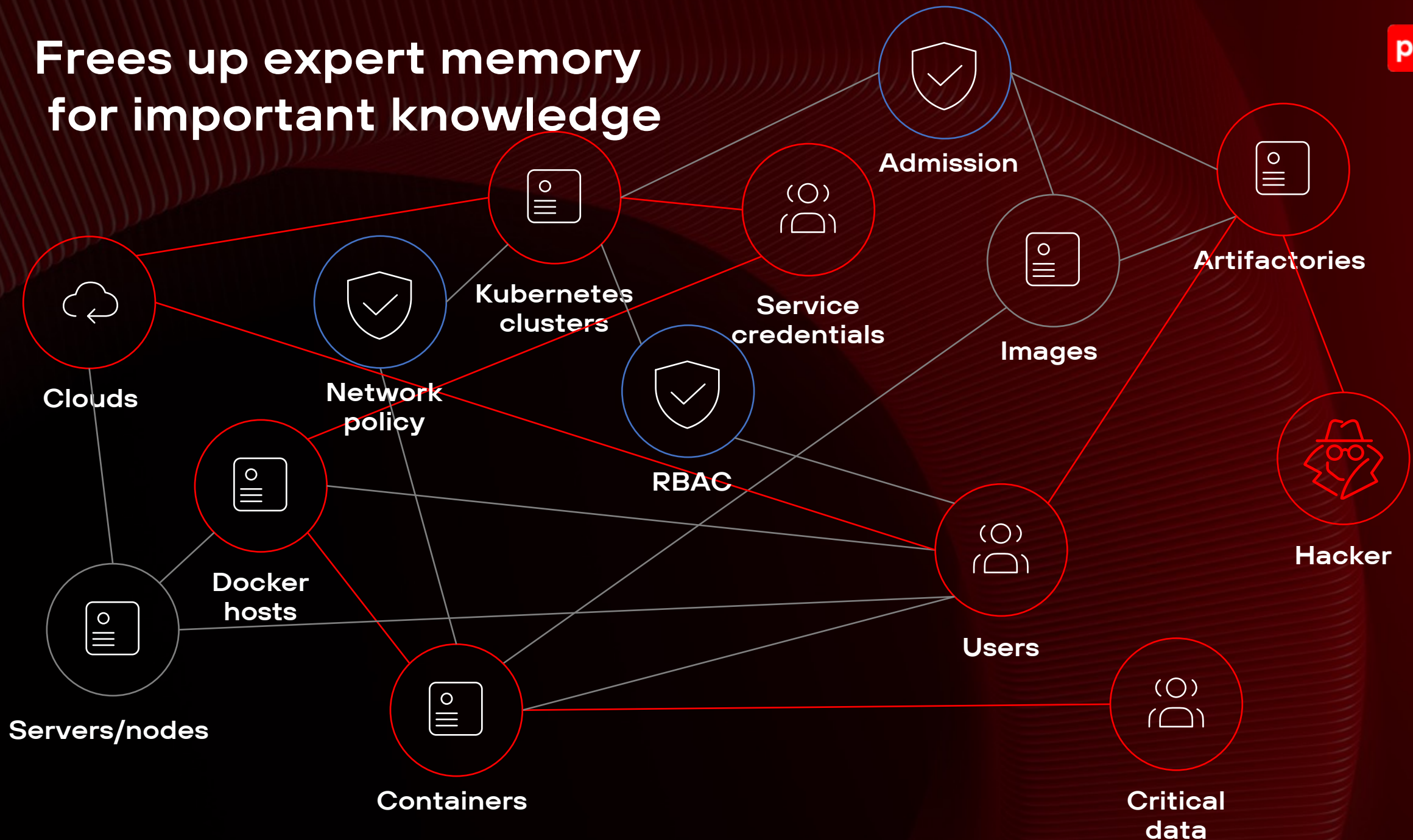
Digital knowledge store of container infrastructure security for experts

- Rules, signatures, and profiles for containers from leaders in result-oriented security.
- CS tunes policies based on user input (removes errors, and suggests to expand monitoring, install new agents, and configure Kubernetes).
 - Provides analytics on container security, where to make fixes, and where to disable.

Container infrastructure security always on hand

- Compatible with all popular runtime types (Kubernetes, Rancher, OpenShift, Deckhouse, Orion Nova, Shturval, Docker, Podman, and Swarm).
- Implements ChatOps and uses classical approaches for security monitoring, reduces response time.
- Built-in policies start working day 0 and show results.

Frees up expert memory for important knowledge



Smart expert and assistant

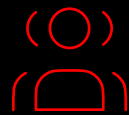


It's important for our users that the system provides recommendations on settings and relevant container security content based on an analysis of their infrastructure. It also needs to visualize potential threats and explain their context. Support from the community and vendor are crucial.



PT Essential

Rules and standards from the leader of result-driven cybersecurity



PT CS GPT

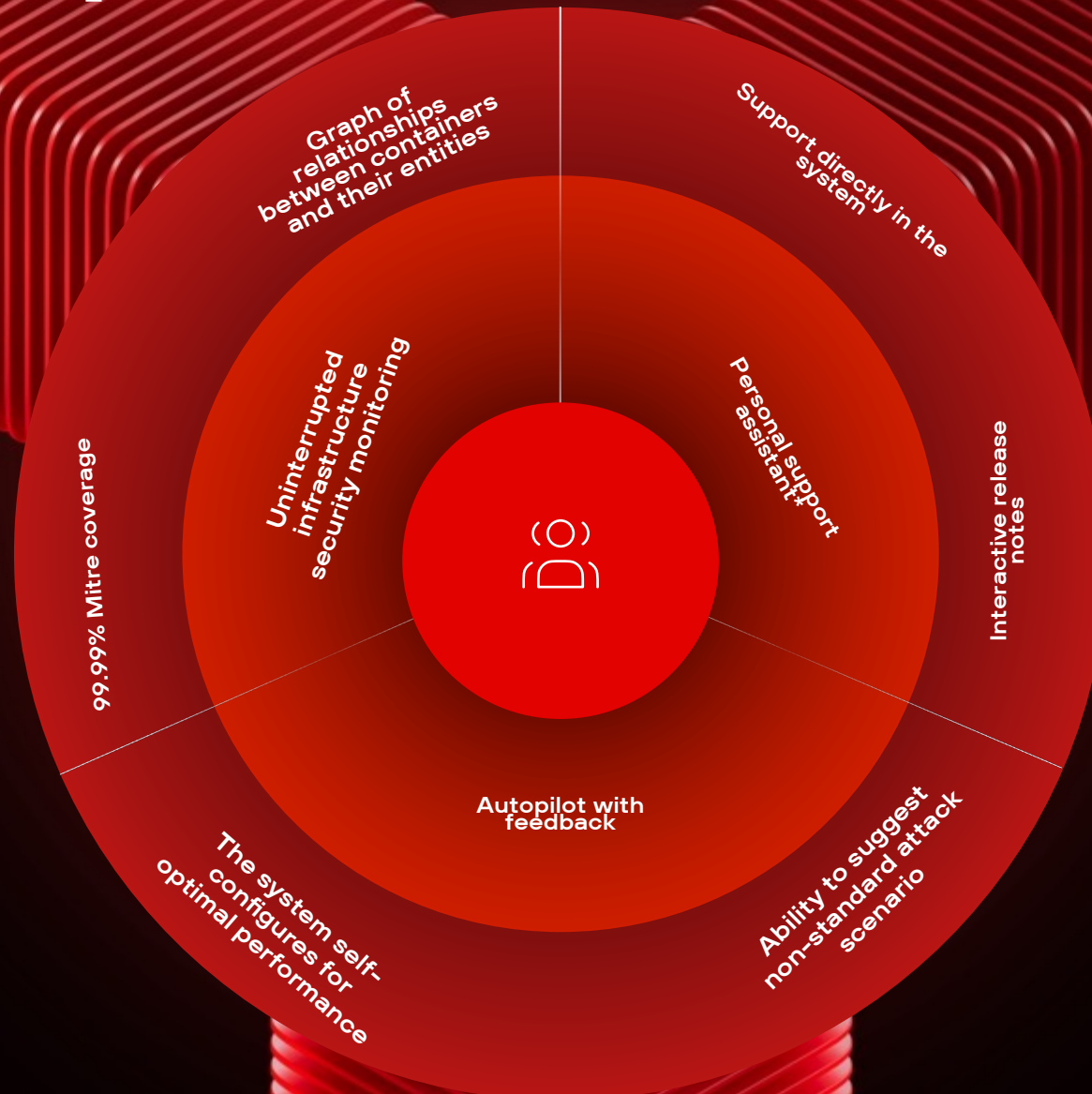
Recommends which service to scale and policies to configure



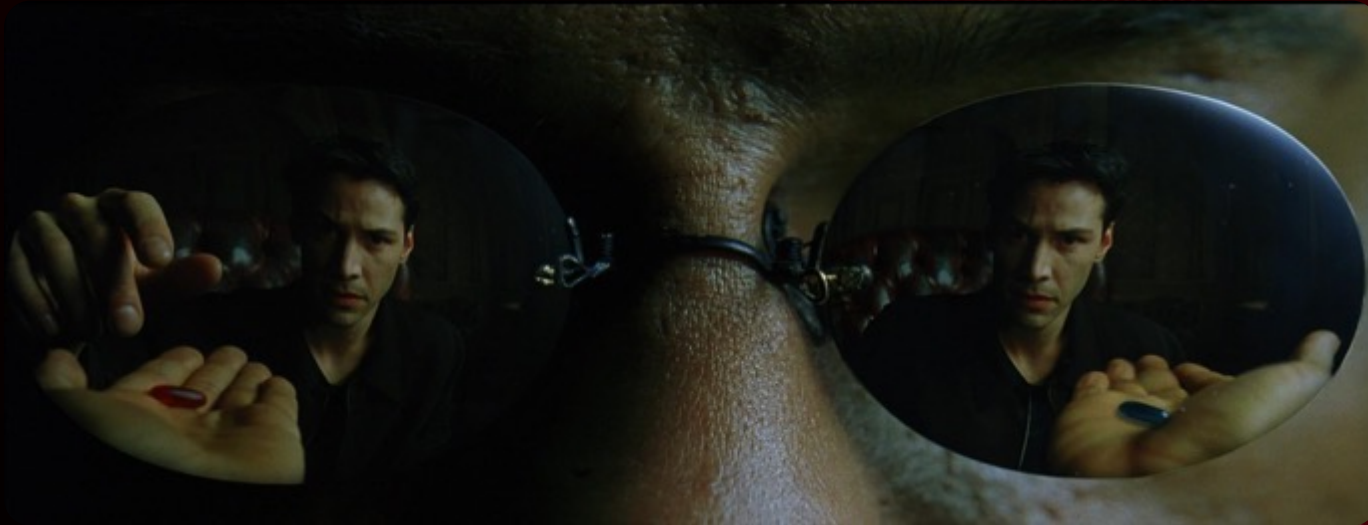
Feedback

Ask the system questions or contact support

Smart Expert and assistant



Its own Smart Expert and assistant

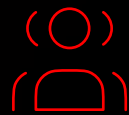


It's important for our users to be able to deeply customize and "file down" each functional module for integration into their security processes. Creating your own content to identify threats and attacks in containers is essential.



Custom content

Full implementation of security-as-code.
Option to embed on the fly or share with others



Attack simulator

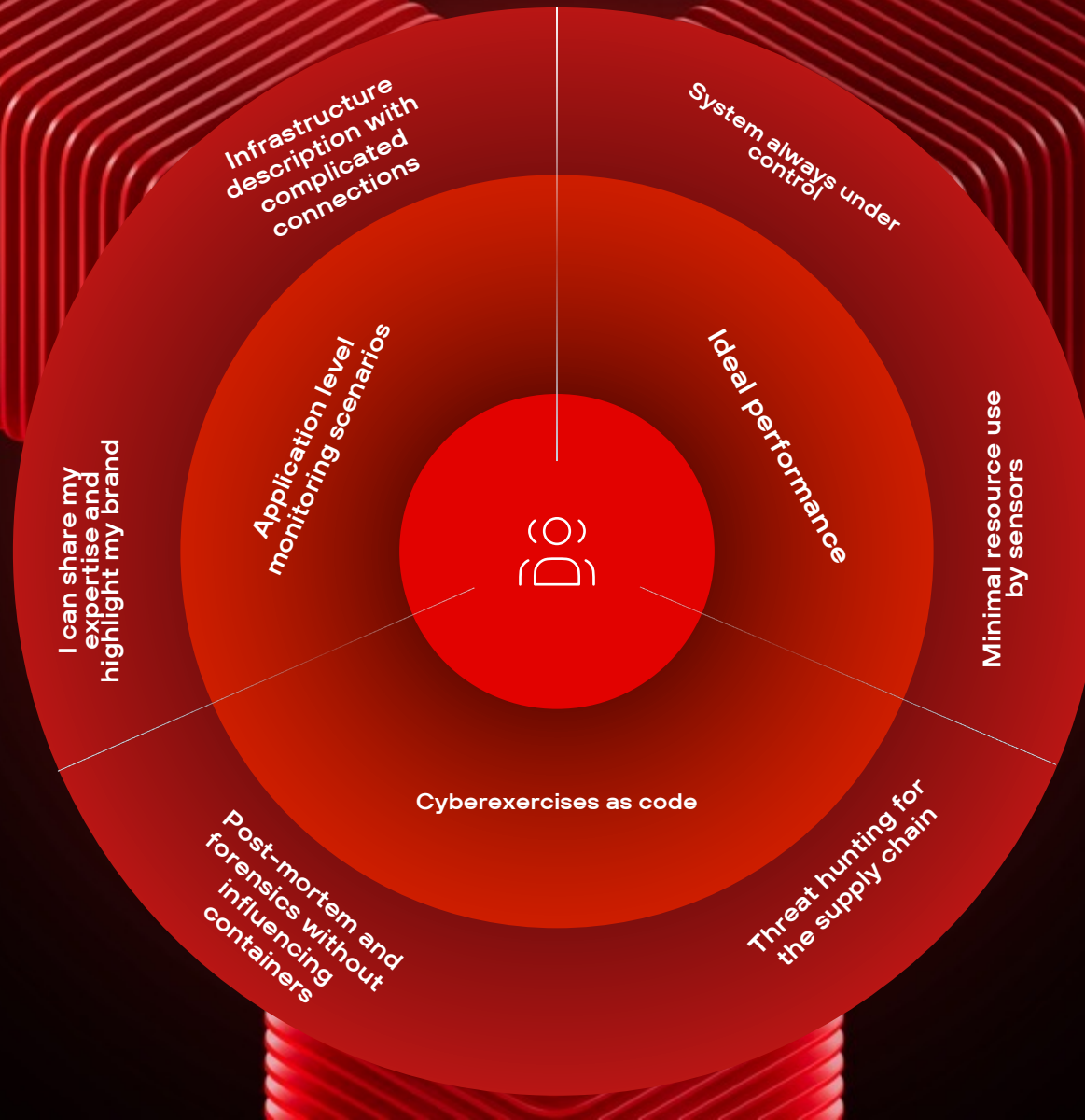
Option to run interactive scenarios and test attack hypotheses



Manual tuning

Each microservice can be scaled and performance tuned, down to the configuration of kernel component monitoring

Its own Smart Expert and assistant



PT CS on the job



PT CS for IT

- Inspection of all images and containers in terms of asset and configuration management (full graph of container infrastructure)
- Observability and alerting for industrial loads (detectors can describe information security events and availability cases)
- Automation of change management related to container infrastructure management (updating images and related configurations)

Easy entry into the product and container security

- SaaS PT CS for experimenters focused on testing hypotheses rather than configuring infrastructure. (Installed agents from a script and logged in to the cloud)
- PT CS Community Edition for enthusiasts ready to share their expertise with the world (quick and easy to install and test content)
- PT CS Online, an interactive demo of the product with case analysis (Standoff) (Cases with varying levels of complexity to hone skills)
- One-click installer for all deployment options

How it works

PT CS features



Management of image vulnerabilities

- Image scans in CI
- Image register scans
- Own database of vulnerabilities
- Monitoring the use of trusted images



Runtime container security

- Monitoring activity in containers
- Anomaly detection
- Visualization of interactions
- Blocking unwanted operations



Clusterization cluster security

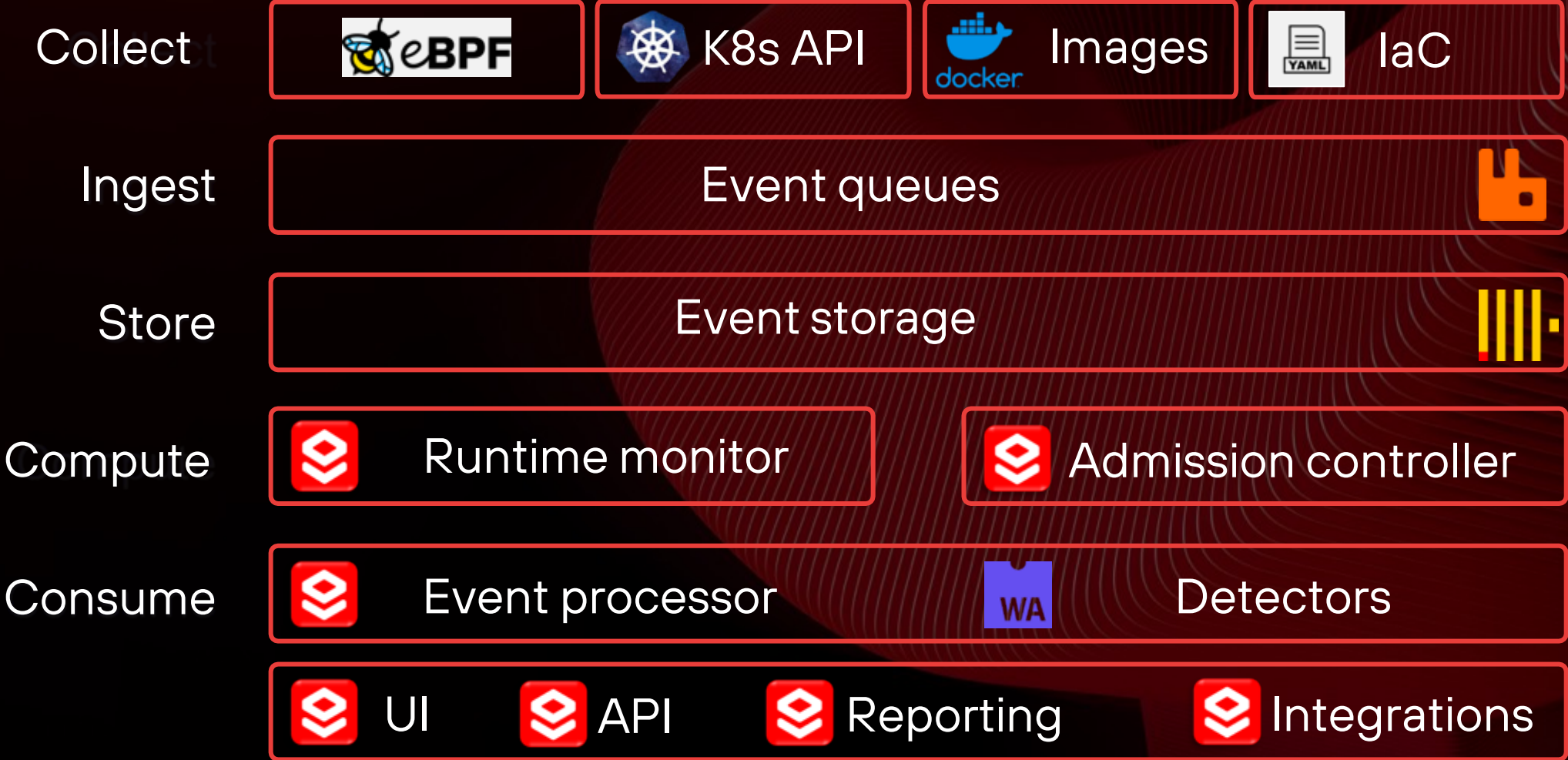
- Protecting clusters from unauthorized configuration changes
- Role-based access control (RBAC)
- Monitoring network policies
- Admission control
- Cluster and node benchmark



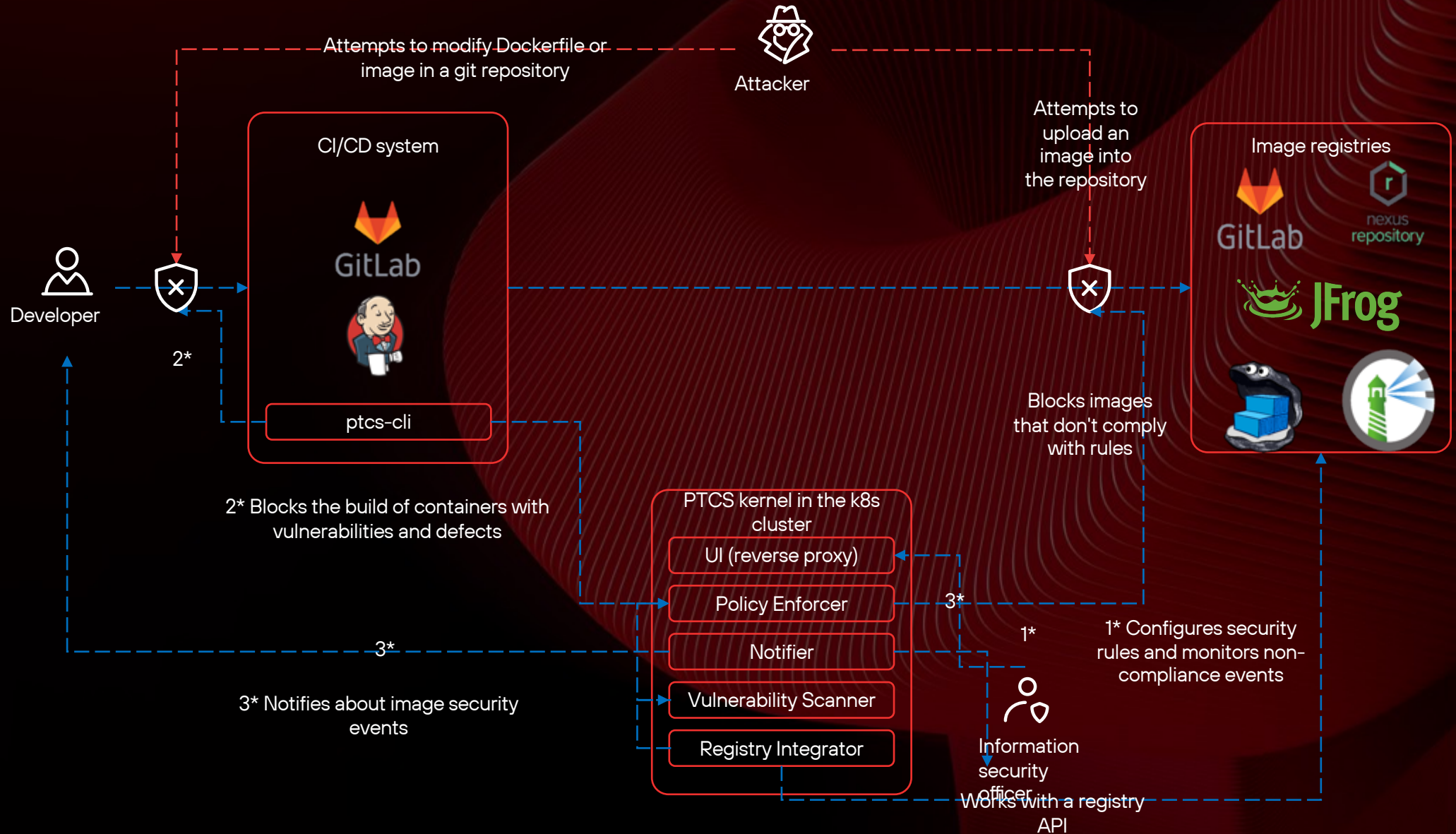
Integration with DevSecOps

- Integration with Application Inspector SAST/DAST
- Integration with information security monitoring tools
- Integration with collaborative development tools

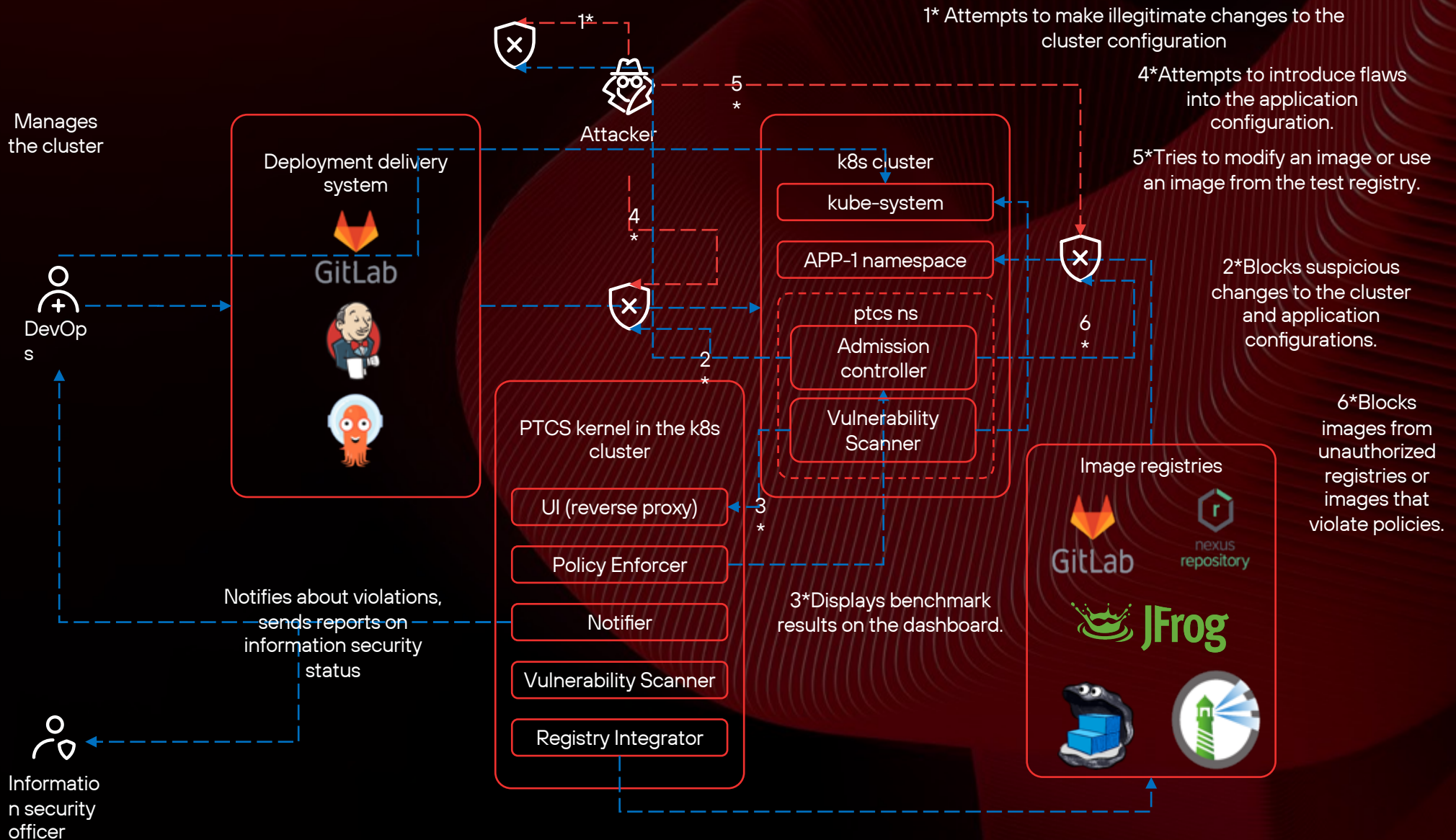
Data we collect



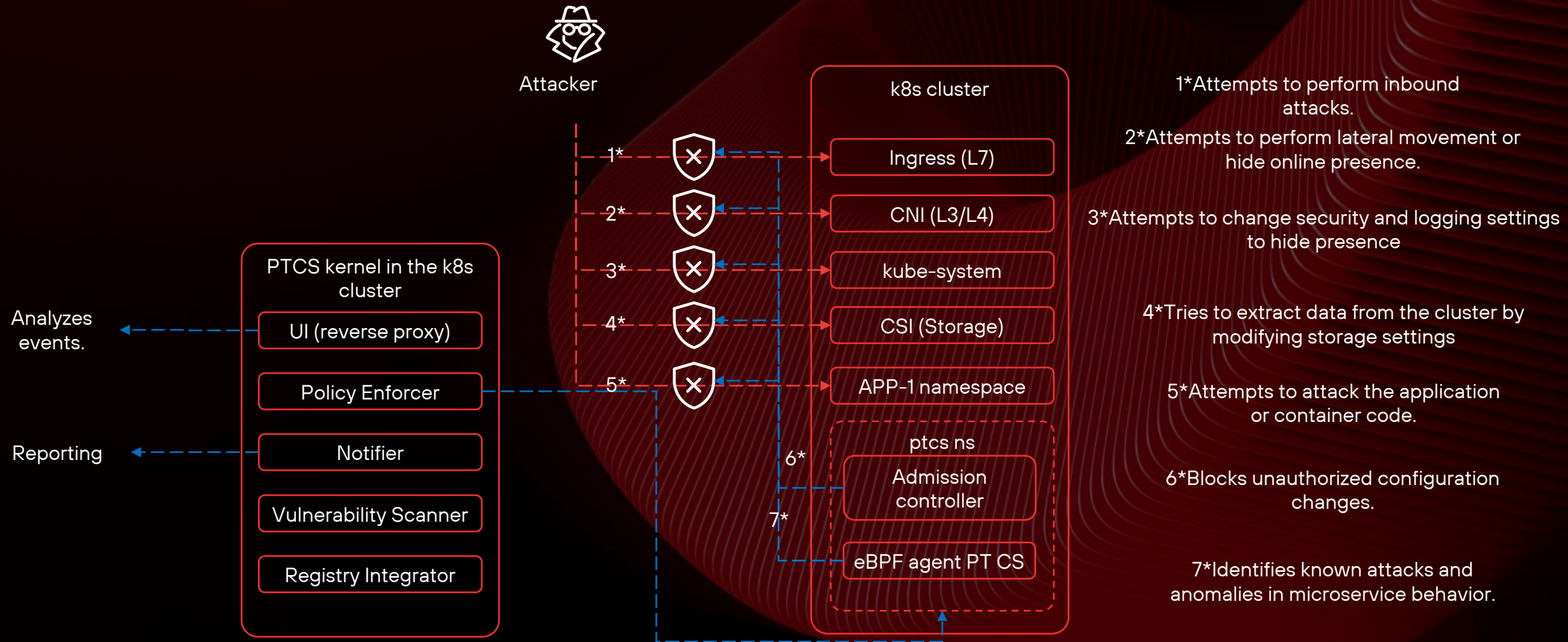
How PT CS works: build stages



How PT CS works: deployment stage



How PT CS works: runtime stage (ind.)



Product roadmap

PT Container Security milestones



Best runtime monitoring

- Content development to enable event monitoring in a container
- Deep tuning of monitoring policies

Single container security console

- Coverage of orchestrators
- Coverage of Docker/Podman
- Everything in one database (multitenancy)

CISO's "second brain"

- Aggregation and automation of best practices for result-driven cybersecurity in containers
- Reports and visualization of relationships between containers and their properties

Autopilot for container security

- Use of LLM when interacting (configuring, searching, tuning) the system
- Smart scaling of only the services you need

Personal expert and assistant

- Additional training and advanced recommendations in the system
- Use of ML and graph algorithms to hypothesize attacks and find weak points in the infrastructure

Database contents by end of 2025



Full cluster coverage

We consider all resources from the cluster and popular CRDs (custom)
We check them according to our standard
We make basic visualizations (even without connections)
Benchmarks as a dashboard and separate report

More than Kubernetes

There are lots of Docker clients (sometimes podman) that objectively don't need kube, the following needs to be covered:

- Runtime monitoring
- API (socket) protection

Reporting and database search

We're learning to search the entire system
Option to generate summary reports on the system
PT CS audits as separate events
(option to download manually)

Multitenancy

- ▣ Agents and scanners will be moved as separate components into external (protected) clusters
- ▣ Option to manage and synchronize settings
- ▣ Reworking the approach to access sharing in the system

What's next?



Queue 1

Queue 2

Queue 3

Expertise

- Reach 100% Mitre coverage for containers.
- Implement anomaly monitoring (white lists)

- Implement customization of the matrix and tag rules so users can see infrastructure coverage.

- Model of recommendations to enable rules
- Tools for the system to assess content quality:
 - Performance
 - Quality

Performance

- We're creating and putting metrics all in one place, and analyzing how the load affects different components.

- Performance guides
- We're adding rules to the system and creating our own alerts so users understand why and when things may get bad.

- Actualize autoscaling to implement PT CS as SaaS (our resource consumption depends and is tuned based on the load).
- Model based on consumption forecast.

Integrations

- Add templates for the most popular:
 - Jira, YT, Redmine
 - DefectDojo, ASOC
 - MP SIEM, Security Vision, R-Vision, O2

- SSO, OIDC, SAML
- Storage of secrets
- Data backup

- New protocols and templates:
 - Hadoop
 - REST API
 - BI - systems

Recommendations

- Show user connections between container objects and highlight dangerous ones (for example, image-container, container-external network, role-token)

- Complex correlations and graphs to identify non-obvious cases (for example, identical accounts in different clusters or nodes)

- ML model that learns from the connections and incidents we show it, and will alert without profiles and signatures



pt@ptsecurity.com



ptsecurity.com