Security Operations Center Workshop

David Crooks

Liviu Vâlsan

Overview

- Background and motivation
- Introduction to the WLCG SOC WG
  - SOC initial model
- PocketSOC
- Plan for the workshop
- Hands on exercises

Background: EGI CSIRT

- EGI CSIRT coordinates operational security activities within the EGI Infrastructure
- Delivers a secure and stable infrastructure
- Gives scientists and researchers the protection and confidence they require to safely and effectively carry out their research

EGI CSIRT

Background: WISE

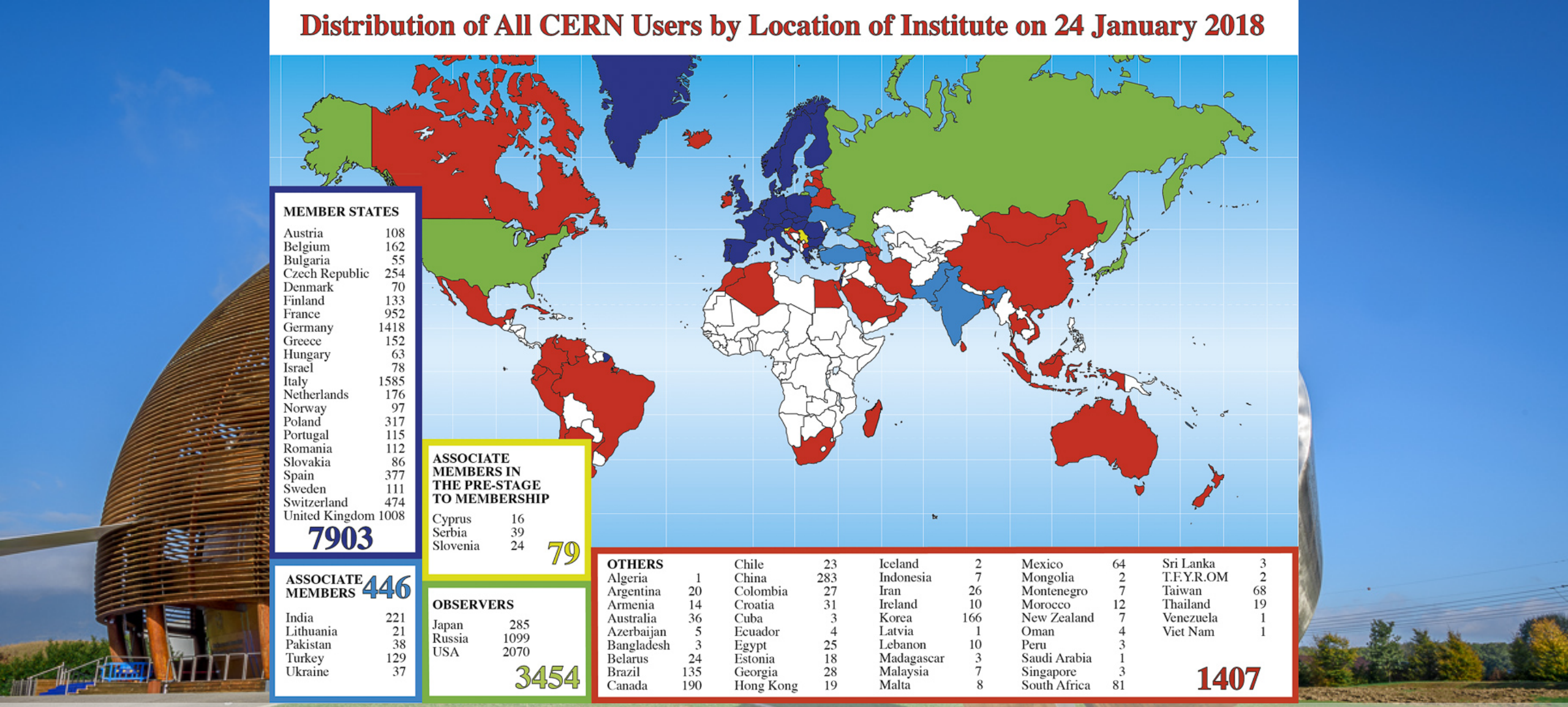
- WISE fosters a collaborative community of security experts and builds trust between IT infrastructures
- Includes cyberinfrastructures, e-infrastructures and research infrastructures
- Experts participate in the joint development of policy frameworks, guidelines, and templates

Founded in 1954

Science for Peace

CERN is the European Laboratory for Particle Physics

Intergovernmental organisation straddling the Franco-Swiss border near Geneva



CERN has 23 member states and supports a global community of 18,000 scientists, 110+ nationalities

Science:
Fundamental research in particle physics

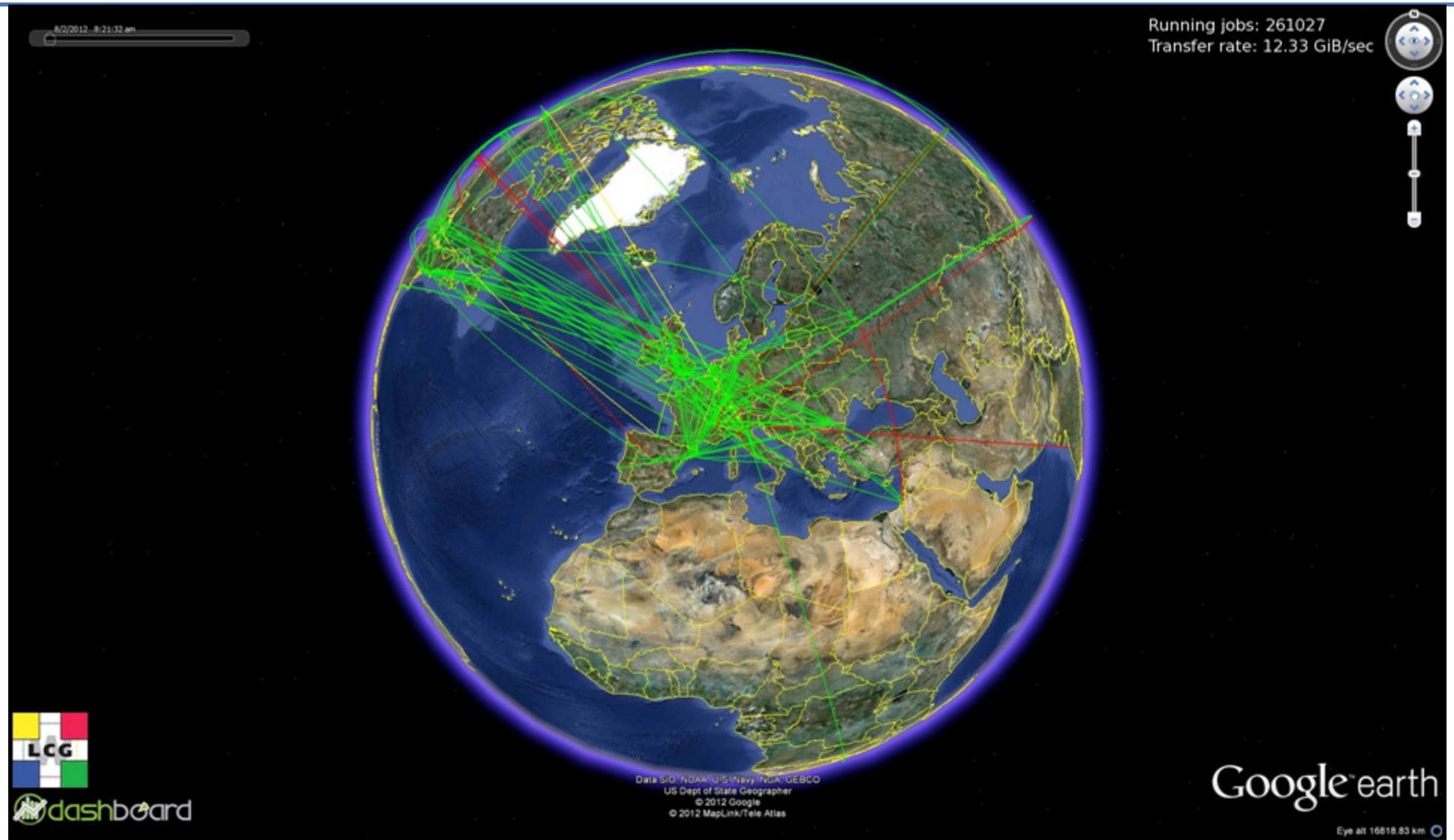
Technology and innovation:
World Wide Web, medical applications

Training and education:
Numerous training programmes

27 km circumference
80 - 140 m underground

The Large Hadron Collider (LHC)

The Worldwide LHC Computing Grid

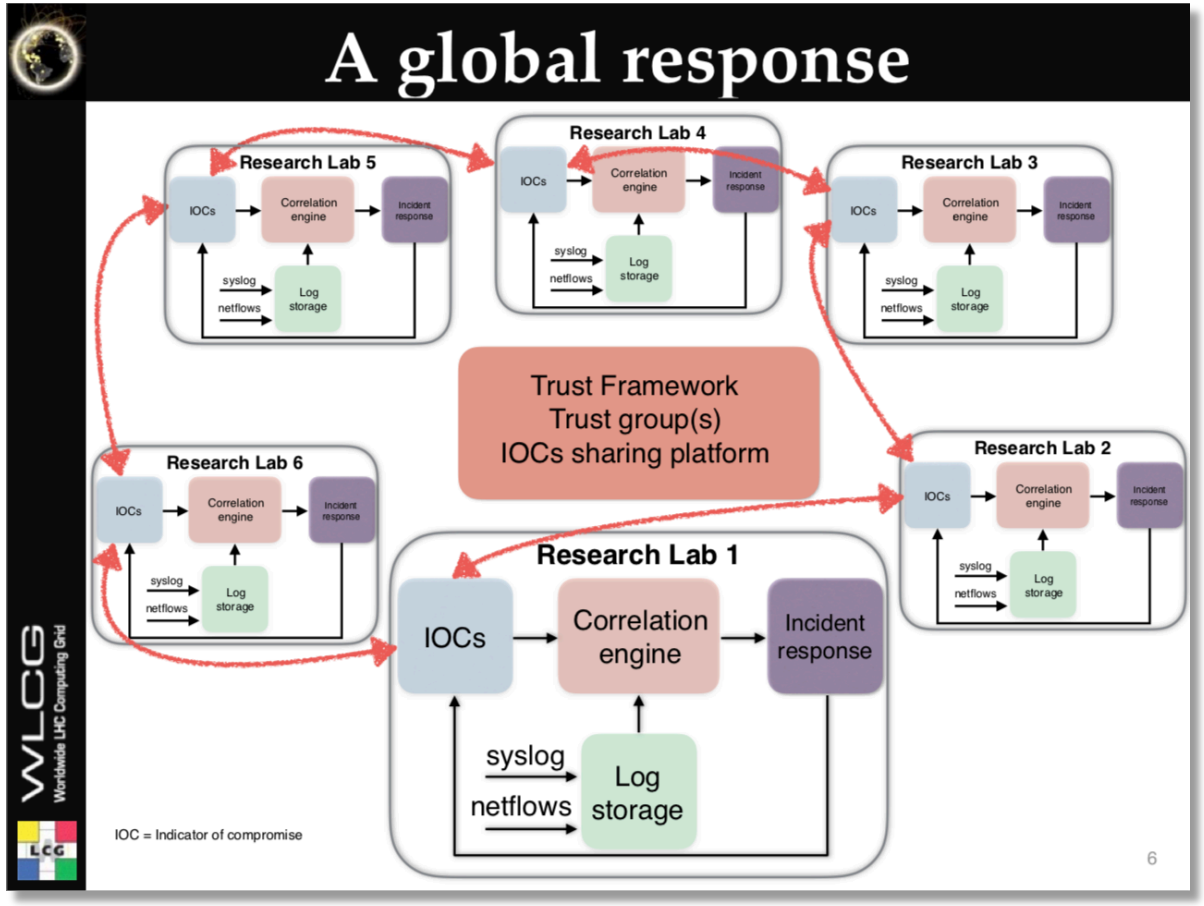


The Worldwide LHC Computing Grid

- Key tool for particle physics research
  - Provides near real-time access to LHC data
- Global collaboration
  - 42 countries
  - 170 computing centres (sites)
  - Over 2 million tasks daily
  - 1 million computer cores
  - 1 exabyte of storage

Motivation

- Tackling modern cyberthreats together is the only way forward
- Romain Wartel, WLCG Security Officer
- Computing in High Energy and Nuclear Physics (CHEP) 2019



Motivation

- Adversaries are motivated and well funded
  - Cybercrime
  - Nation state backed attackers
- Malware as a Service (MaaS)
  - Ransomware
  - Banking trojans
- What do we have?
  - Our infosec community

Motivation

- Within a given community (such as the WLCG), we see similar threats from similar actors
- Acting together, we can establish common response mechanisms and support each other
- By sharing threat intelligence we can better inform fellow organisations to take action
  - Active (e.g. firewall/DNS blocks)
  - Passive (e.g. detection, awareness and improved response)

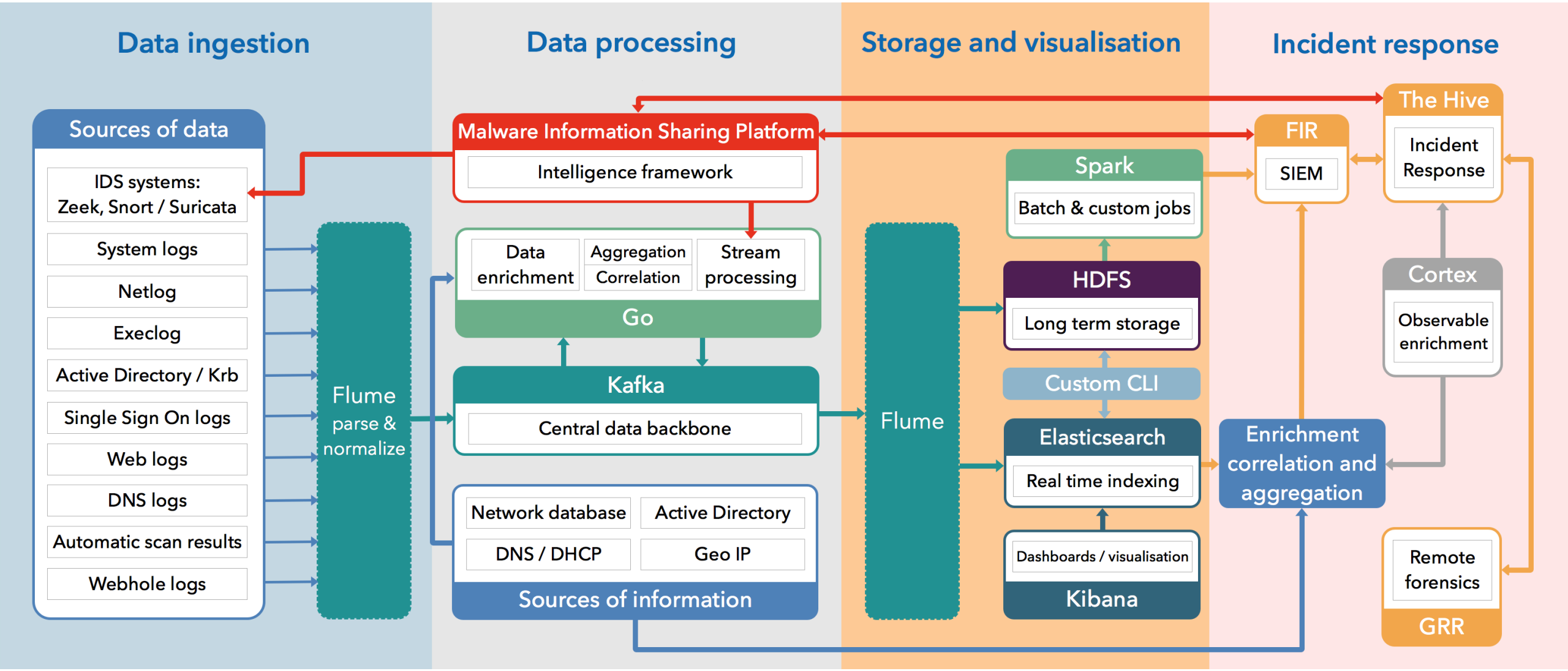
Introduction

- Key element: sharing between trusted parties
  - How to achieve this?
- Some scientific infrastructures already have a level of trust
  - built on operational Memorandums of Understanding
    - organisational trust
  - years of effort

Introduction

- WLCG SOC WG is mandated to create reference designs to allow WLCG sites to
  - Ingest security monitoring data
  - Enrich data, store and visualize
  - Alert based on matches between this security data and threat intelligence (Indicators of Compromise or IoCs)

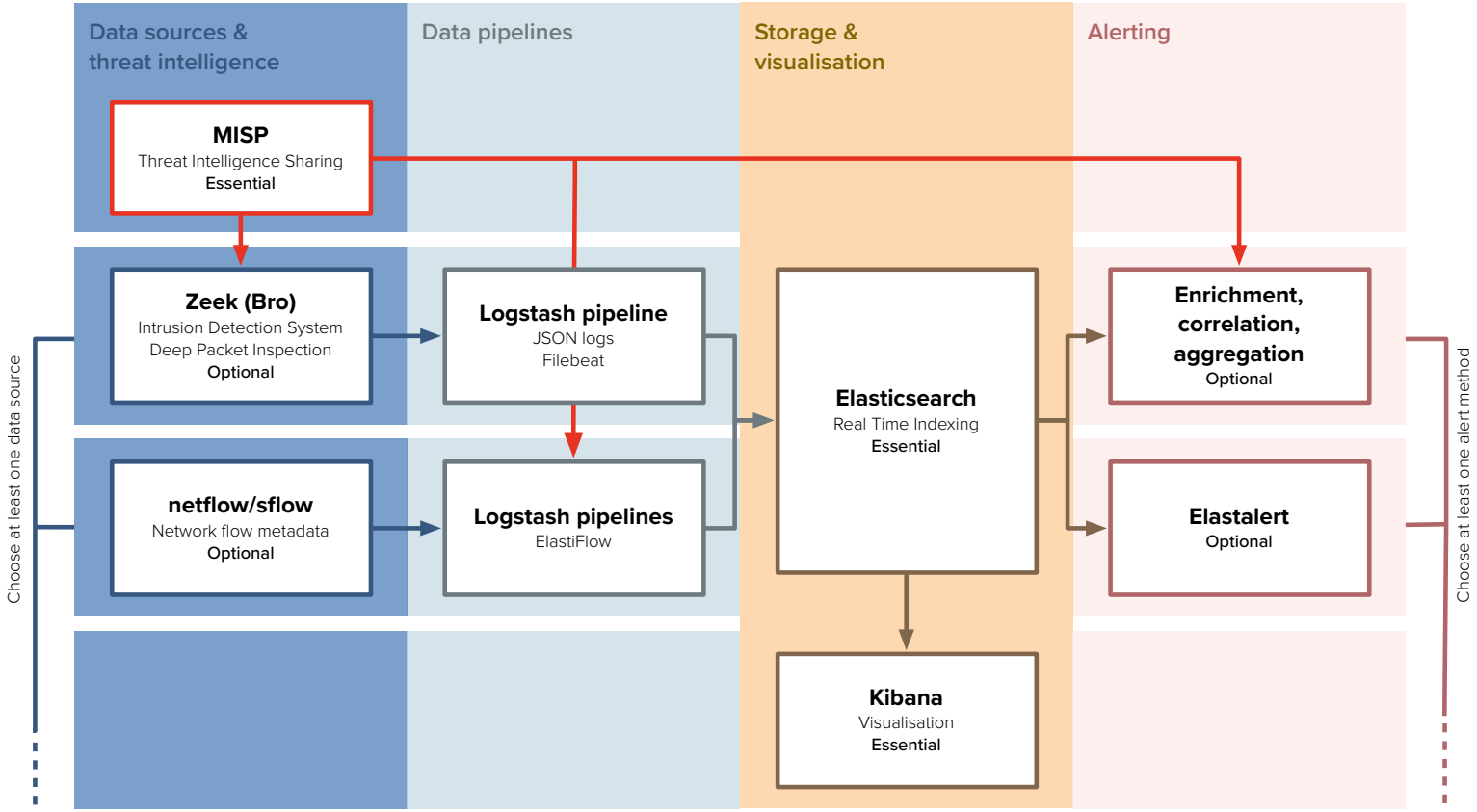
CERN SOC diagram



Roadmap

- Principle of minimum viable product
- Start from most important elements
  - Threat intelligence
  - Network monitoring
  - Storage and visualization
  - Alerting
- Expand to include more capabilities over time

SOC WG Initial Model



Technology stack

Stage	Component	Notes
Threat intelligence	MISP	Cornerstone of model; focused around central MISP instance hosted at CERN
Data sources	Zeek	Highly detailed but requires dedicated hardware
	Netflow	Readily available at many sites but offers less information than Zeek
Data pipelines	Logstash + Filebeat + JSON logs (e.g. Zeek)	Basic pipeline provided by WG
	Logstash + Elastiflow (Netflow)	Dedicated pipeline for netflow/sflow
Storage and Visualisation	Elasticsearch	Share deployment configs within group
	Kibana	Share dashboard processes
Alerting	Correlation scripts	Generalised version of CERN scripts
	Elastalert	Rule based alerts; share typical configs

Academic MISP instance

- Initial sharing model
  - Hub and spoke
  - Benefit from CERN trust relationships and experience
- Mostly TLP:GREEN and TLP:WHITE
- TLP:AMBER events produced by CERN

Academic MISP instance

- Start with sites pulling event data from central instance
- Via
  - Web app (visually inspect data, publish new events or contribute to existing events)
  - API client (direct export to IDS)
- For advanced sites possibility to sync from central instance to their local MISP instance

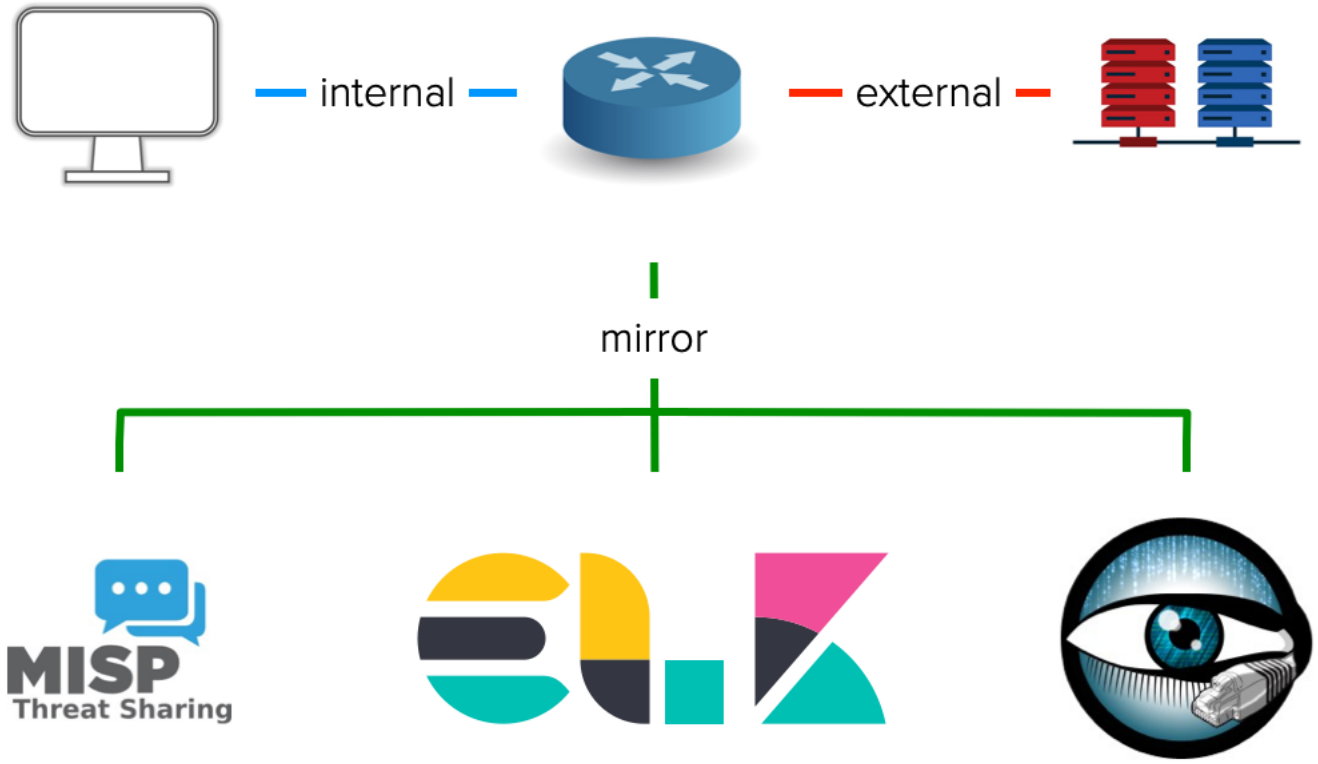
Workshop introduction

- Up until now we've been setting the stage
- Agenda and goals for today
  - Design requirements
    - This presentation!
  - Demonstration of Dockerised version of the initial model
    - PocketSOC
  - Range of exercises to explore capabilities of SOC model
    - Guided by experience and interest in the room

PocketSOC

- PocketSOC is intended as
  - Demonstration of the individual components in the SOC initial model
  - Testbed for adding new components (recently Elastiflow)
- Docker cluster, orchestrated with docker-compose
- Intended as tool for training events to demonstrate capabilities

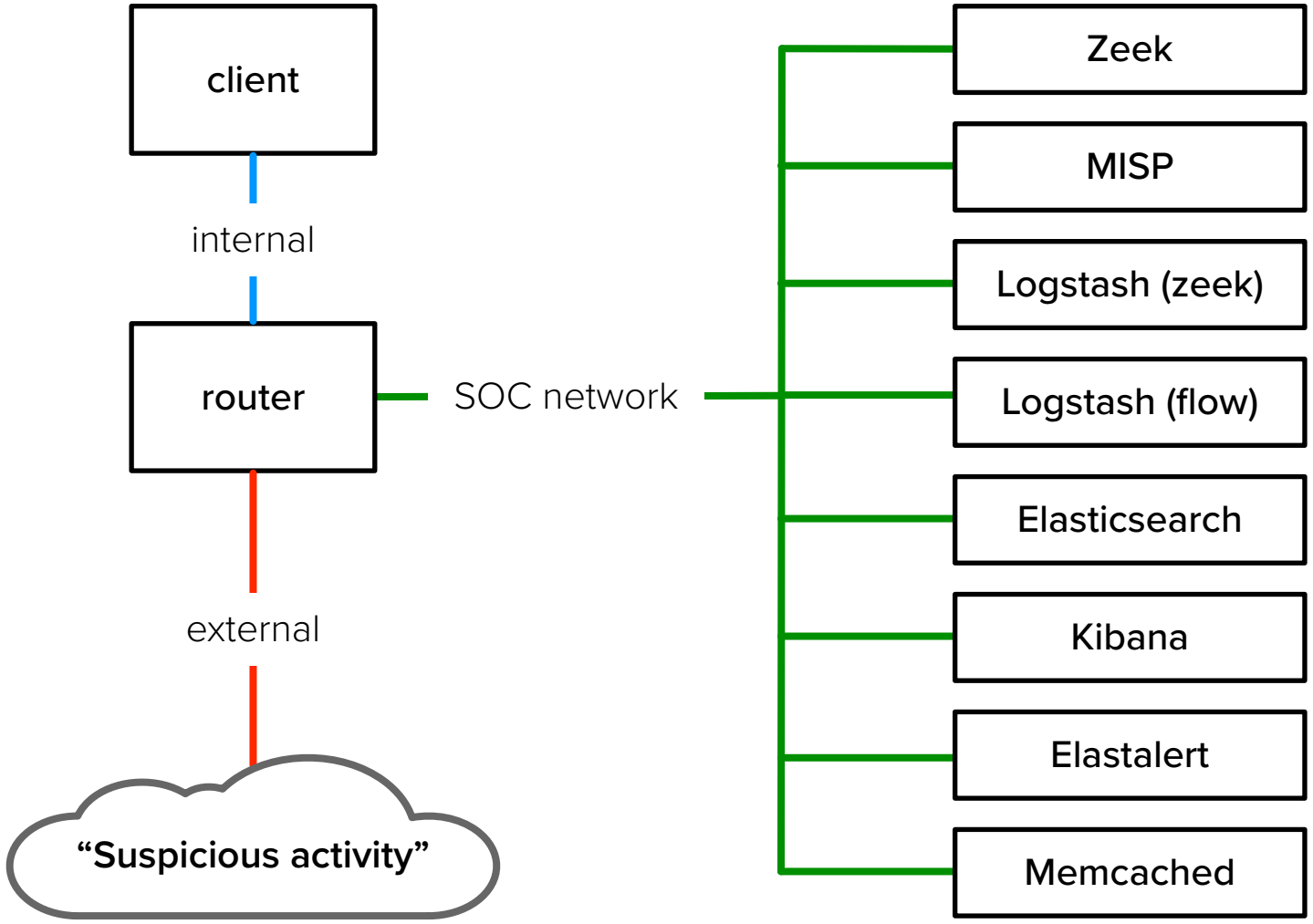
PocketSOC block diagram



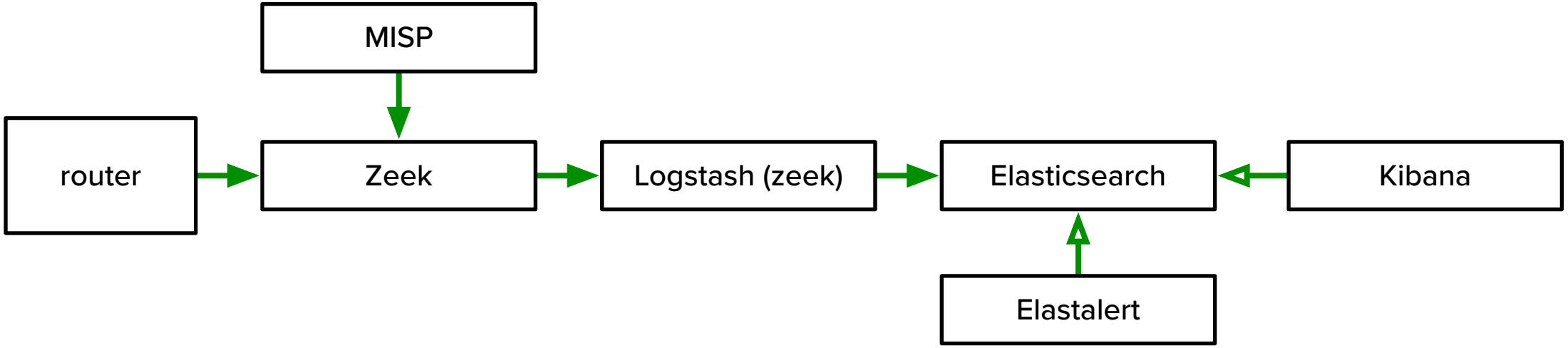
PocketSOC networks

- Internal
  - Client and Router only
- External
  - Router and “suspicious activity”
- SOC network
  - Router and all SOC components

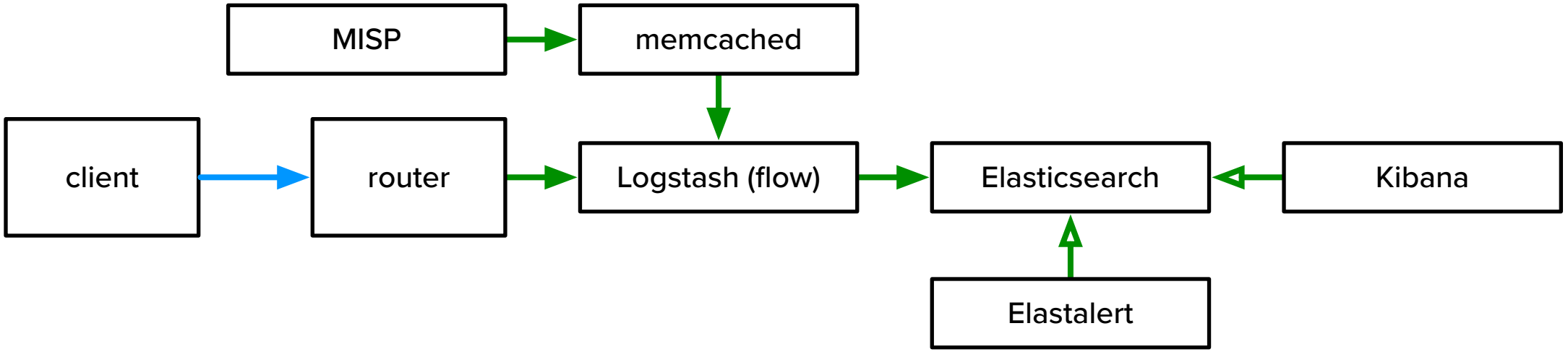
PocketSOC


client
internal
router
SOC network
external
“Suspicious activity”
Zeek
MISP
Logstash (zeek)
Logstash (flow)
Elasticsearch
Kibana
Elastalert
Memcached

Zeek data path



Netflow data path



Scenario

- Suspicious activity is triggered on client
- Check that traffic is observed by Zeek and Netflow
- Generate MISP event based on intelligence
- Check that IoCs are exported to Zeek and Memcache
- [Optional] Configure alerting and see alerts based on suspicious traffic

Preparation

1. Download VM: http://cern.ch/go/xzL6
   - We have some USB sticks available
2. Instructions: http://cern.ch/go/plb7
   - A snapshot of these is in `/opt/pocketsoc/README.md`
3. The VM is preconfigured with 6GB of RAM
   - If possible, 8GB of RAM is preferred
4. Credentials
   - `pocketsoc:pocketsoc`
   - `ssh` to `localhost:2222`
5. `sudo -s`

Exercise 1: Trigger activity

- Steps within PocketSOC to trigger activity
  - `pocketsoc_attach client`
  - On client, `/files/suspicious_activity.sh`

Exercise 1: Zeek logs

- Look for evidence of traffic in zeek logs
  - Use Kibana in Discover tab
  - Use `zeek-*` index pattern
    - Should be the default

Exercise 1: Elastiflow logs

- Examine Elastiflow logs when available
  - Discover tab
    - `elastiflow-*` index pattern
  - Download the dashboard file
    - https://raw.githubusercontent.com/robcowart/elastiflow/master/kibana/elastiflow.kibana.7.5.x.ndjson
  - Visit Kibana Management
    - http://localhost:8080/app/kibana#/management/kibana/objects?_g=()
  - Click "Import" and follow instructions

Exercise 2: IoCs

- Have information by email that `172.19.0.77` has been flagged as malicious
  - E.g. associated with a bot that is regularly contacting C2 server
  - (Ideally have existing MISP sync in place)

- Create a matching MISP event

Exercise 2: Create MISP event

- Create event with appropriate metadata
  - Name, TLP tag, etc...
- Within event create:
  - Object for file download
    - Filehash: MD5, SHA1
  - IP:port attribute
  - URL attribute
- Check that all attributes are marked for IDS
- Publish event

Exercise 3: Check IoC export

- Check that IoCs are being exported from MISP
- Into Zeek
  - `pocketsoc_zeek_intel`
  - Checks contents of zeek intelligence feed
- Into Memcached (enriches elastiflow pipeline)
  - `pocketsoc_memcache`
  - Dumps contents of memcache
  - Only IP addresses: other IoC types not relevant for netflow

Exercise 4: Alerting

- In PocketSOC Elastalert is configured with two rules
  - `misp_src`: checks for matches to MISP events in netflow data
  - `zeek-intel`: checks for zeek-intel records, indicating results found by Zeek intelligence framework

- Out of the box, configured to alert to Telegram

Exercise 4: Telegram alerting

1. Create Telegram bot and note the token
   - For example follow screenshots here
2. Create New Group and add bot
3. Send message to group
   - If necessary, disable privacy mode for the bot
4. Check API to get group id
   - https://api.telegram.org/bot[BOTTOKEN]/getUpdates
   - Want chat.id: “-xxxxxxxxxx”
5. Provision credentials with `pocketsoc_elastalert_secrets`

Exercise 4: Other alerting

- Many other alerting options are possible
  - Command, Email, JIRA, OpsGenie, SNS, HipChat, Slack, GoogleChat, Debug, Stomp, theHive
- Refer to https://elastalert.readthedocs.io
  - Exercise for reader ☺

Exercise 4: Check alerts

- Check alerts are received via `misp_src` and `zeek_intel`
  - May need to retrigger activity
  - `misp_src` may take longer to trigger depending on when netflow records become available

Other resources

- Where to go for more training?
  - Including but not limited to…
- MISP
  - Regular MISP events
    - https://circl.lu
  - Online documentation and training
    - https://www.misp-project.org/documentation/
    - https://github.com/MISP/misp-training

Other resources

- Zeek
  - Regular Zeek events
    - https://zeek.org
  - Online documentation and training
    - https://www.zeek.org/documentation/index.html
    - https://www.zeek.org/documentation/tutorials/index.html
    - https://www.zeek.org/documentation/slides/index.html
    - https://try.zeek.org/#/?example=hello

Other resources

- Elasticsearch, Logstash and Kibana
  - https://www.elastic.co
- Elastalert
  - https://elastalert.readthedocs.io/en/latest/
- Elastiflow
  - https://github.com/robcowart/elastiflow

Other resources

- PocketSOC
  - https://gitlab.cern.ch/wlcg-soc-wg/PocketSOC
  - (moving to Github in due course)

Contact details

- David Crooks (david.crooks@stfc.ac.uk)
- Liviu Vâlsan (liviu.valsan@cern.ch)
- WLCG SOC WG
  - https://wlcg-soc-wg.web.cern.ch/
- PocketSOC
  - https://gitlab.cern.ch/wlcg-soc-wg/PocketSOC