Evolving SVG

Linda Cornwall, STFC, UKRI

UK HEPSYSMAN May 23rd 2019

eosc-hub.eu

@EOSC_eu

Dissemination level: Public

EOSC-hub receives funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 777536.

What is SVG? (reminder)

- SVG = Software Vulnerability Group
- Main Purpose – to prevent Security Incidents due to software vulnerabilities
  - In EGI
  - But NOT trying to substitute/compete with various other vulnerability activities external to EOSC-hub/EGI
- Been running in current form since 2005 with relatively minor changes including
  - Going from being focussed on Grid Middleware to all types of software on the EGI distributed infrastructure
  - Encompassing EGI FedCloud

EGI SVG basic procedure

- SVG has been handling vulnerabilities since 2005
  - Handling vulnerabilities which affect the EGI infrastructure and its predecessors
  - To help prevent security incidents
- Anyone may report an issue by e-mail to
  report-vulnerability@egi.eu
- If it has not been announced, SVG contacts the software provider and the software provider investigates (with SVG member, reporter, others)
- If relevant to EGI the risk in the EGI environment is assessed, and put in 1 of 4 categories – 'Critical', 'High', 'Moderate' or 'Low'
- If it has not been fixed, Target Date (TD) for resolution is set - 'High' 6 weeks, 'Moderate' 4 months, 'Low' 1 year
- Advisory is issued by SVG
  - If the issue is 'Critical' or 'High' in the EGI infrastructure
  - When the vulnerability is fixed if EGI SVG is the main handler of vulnerabilities for this software, or software is in EGI Repository regardless of the risk.
  - If we think there is a good reason to issue an advisory to the sites.
- Critical vulnerabilities are handled with top priority, aiming for a resolution within 1 day

- https://documents.egi.eu/public/ShowDocument?docid=3145

Why SVG needs to change

- Proliferation of software and technology used has occurred
  - The distributed infrastructure is less and less homogenous as time goes on
- Now including services in the EOSC-hub Catalogue
- Software Vulnerability Group (SVG) Risk Assessment Team (RAT) cannot be experts in all software, services and configuration
  - Nor can they be looking out for advisories on all software that may be used
- Need a new approach
  - SVG has to hook into the evolving world the best way we can

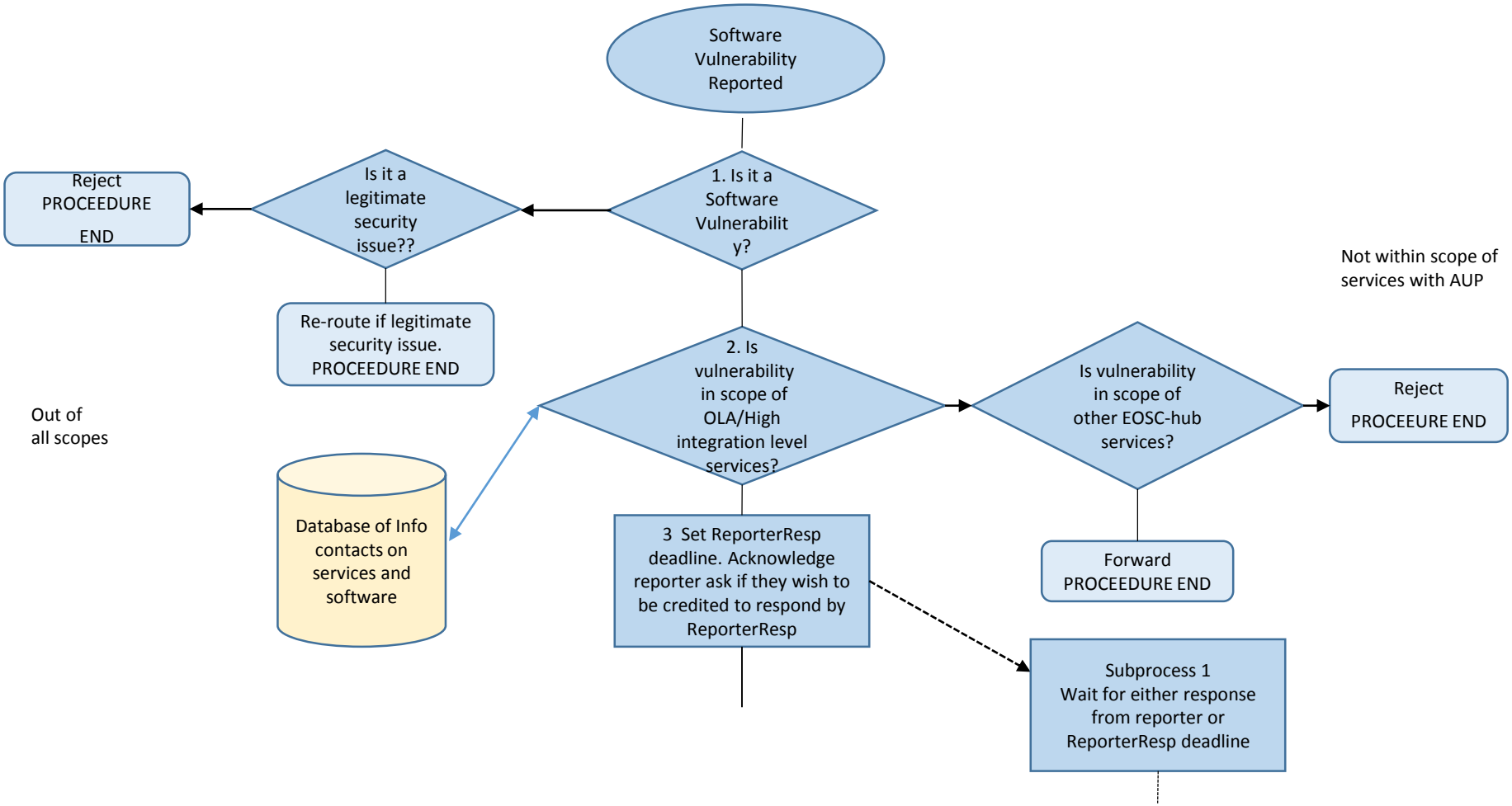
From meetings…

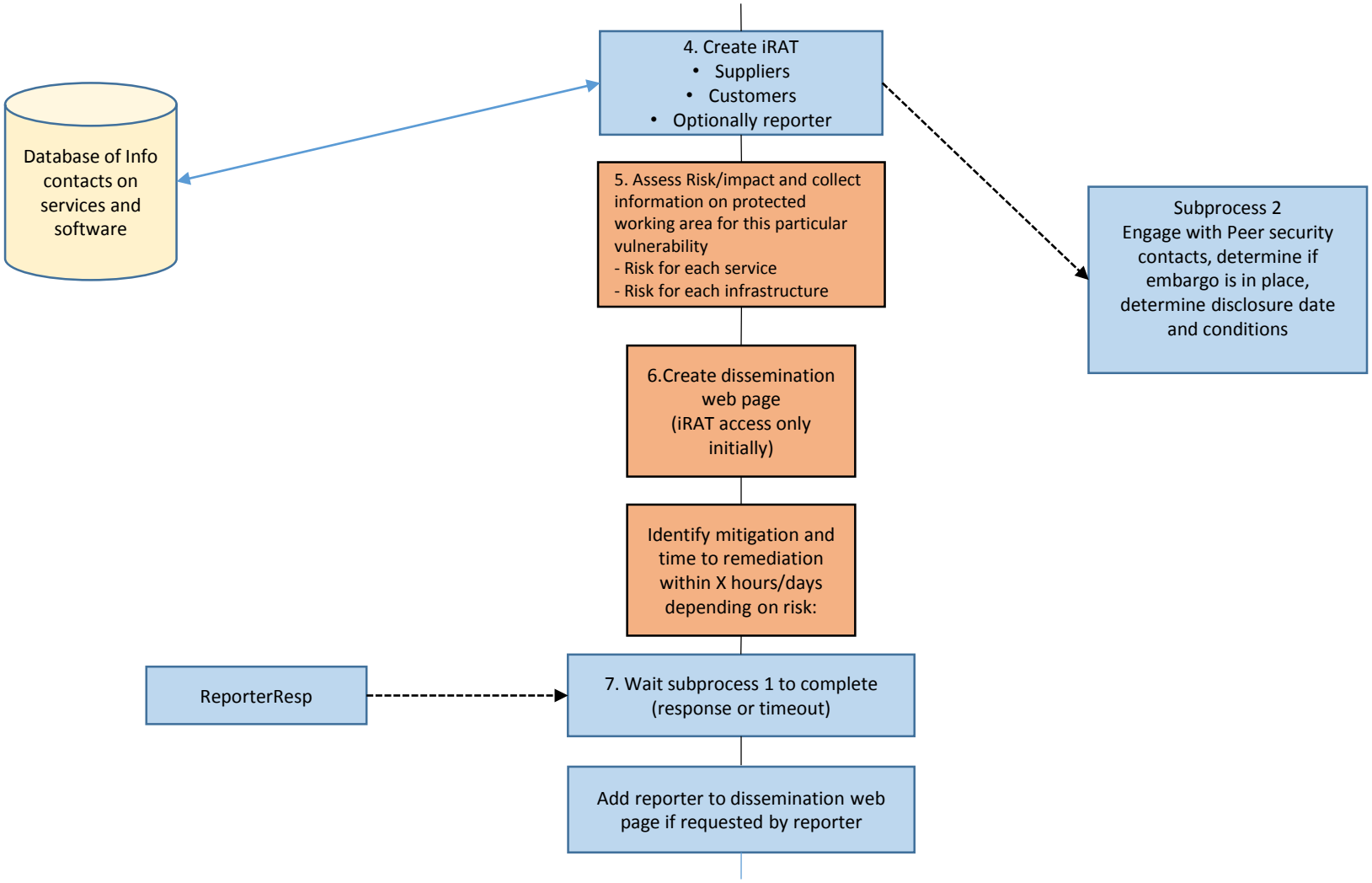
- Many sites are saying ‘we are using X software or Y software in such a way to enable our services’
- Such people need to think about what software they are using, SVG has a checklist to help

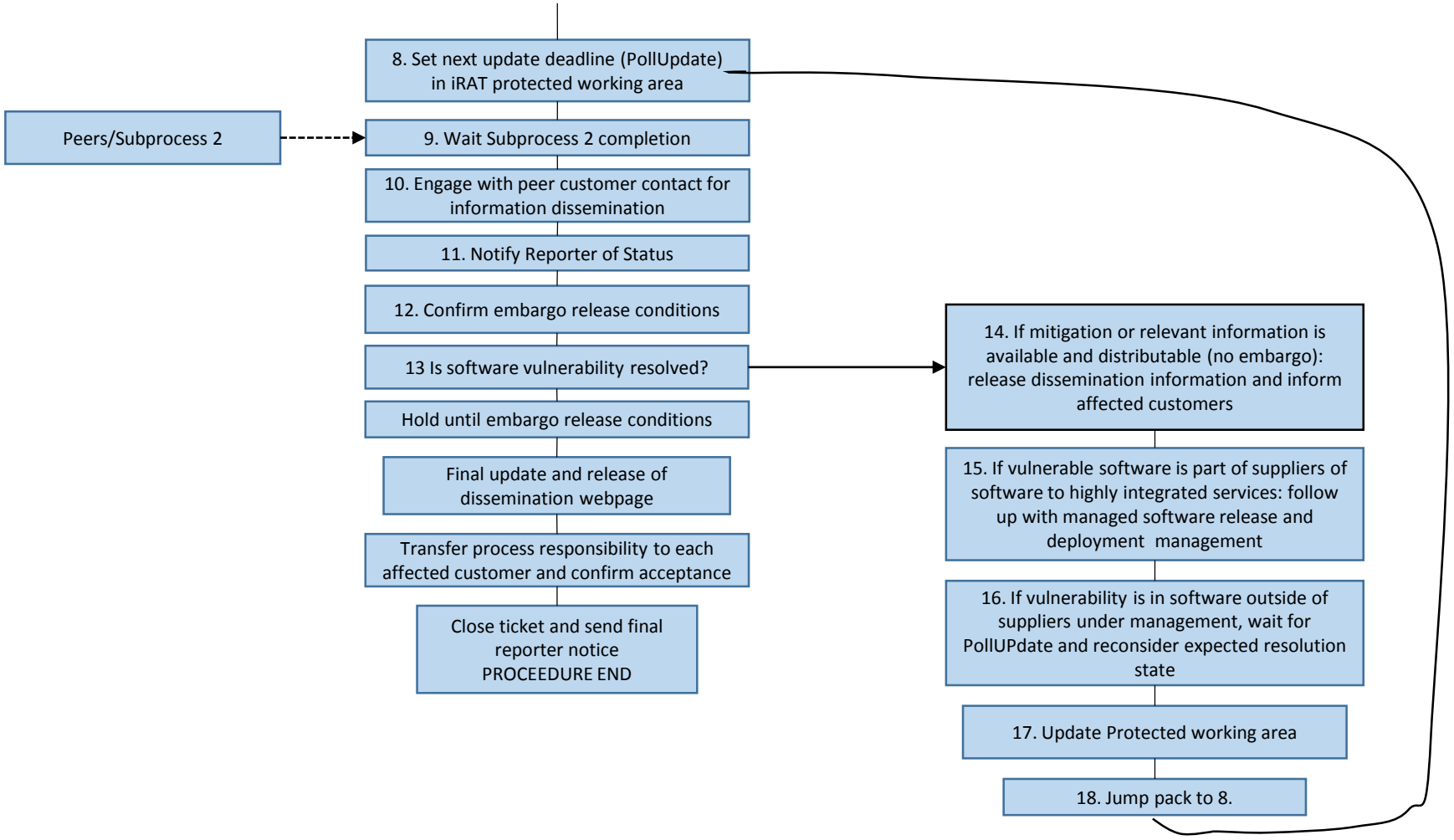
  https://wiki.egi.eu/wiki/SVG:Software_Security_Checklist
- For Services in the EOSC-hub catalogue
  - The person responsible for the service must be a contact, and/or provide contact(s) who know how those services operate and can help investigate relevant vulnerabilities, and look out for vulnerabilities

To move forwards

- Need to depend on experts on software and services to assess a vulnerability
- When a new software vulnerability is reported:-
  - Need to be able to contact the appropriate experts easily
    - Software developers
    - Those who set up services which depend on the software
  - Then set up an Issue Risk Assessment Team (iRAT) to handle this vulnerability
- Devised a new procedure – first in April 2018, then revised in November 2018
  - Put into FitSM format (EOSC-hub using this)
  - And a diagram


Software Vulnerability Reported
1. Is it a Software Vulnerabilit y?
Is it a legitimate security issue??
Reject PROCEEDURE END
Re-route if legitimate security issue. PROCEEDURE END
Not within scope of services with AUP
2. Is vulnerability in scope of OLA/High integration level services?
Is vulnerability in scope of other EOSC-hub services?
Reject PROCEEURE END
Out of all scopes
Database of Info contacts on services and software
3 Set ReporterResp deadline. Acknowledge reporter ask if they wish to be credited to respond by ReporterResp
Forward PROCEEDURE END
Subprocess 1 Wait for either response from reporter or ReporterResp deadline


4. Create iRAT
• Suppliers
• Customers
• Optionally reporter
Database of Info contacts on services and software
5. Assess Risk/impact and collect information on protected working area for this particular vulnerability
- Risk for each service
- Risk for each infrastructure
Subprocess 2
Engage with Peer security contacts, determine if embargo is in place, determine disclosure date and conditions
6.Create dissemination web page (iRAT access only initially)
Identify mitigation and time to remediation within X hours/days depending on risk:
ReporterResp
7. Wait subprocess 1 to complete (response or timeout)
Add reporter to dissemination web page if requested by reporter


8. Set next update deadline (PollUpdate) in iRAT protected working area
Peers/Subprocess 2
9. Wait Subprocess 2 completion
10. Engage with peer customer contact for information dissemination
11. Notify Reporter of Status
12. Confirm embargo release conditions
13 Is software vulnerability resolved?
Hold until embargo release conditions
Final update and release of dissemination webpage
Transfer process responsibility to each affected customer and confirm acceptance
Close ticket and send final reporter notice
PROCEEDURE END
14. If mitigation or relevant information is available and distributable (no embargo): release dissemination information and inform affected customers
15. If vulnerable software is part of suppliers of software to highly integrated services: follow up with managed software release and deployment management
16. If vulnerability is in software outside of suppliers under management, wait for PollUPdate and reconsider expected resolution state
17. Update Protected working area
18. Jump pack to 8.

Main Change

- Those who are responsible for services look out for relevant vulnerabilities and report them
- SVG-RAT then finds all relevant people to form the iRAT.
- iRAT does risk investigation and risk assessment, works out how to mitigate.
- Then advisories on what to do relevant to different services can be made
  - In many cases just update software
- So a consistent risk assessment is provided, advice on how to act is produced, but by the iRAT not SVG-RAT.
- SVG-RAT becomes more of a coordinator, less of an investigator, ensures process runs and there is a consistent approach
- People who are experts in various services help others avoid incidents due to software vulnerabilities via the SVG.

Lots to do!

- We have identified a fair number of tools and sub-procedures to make this work
- But, the most important ones are determining Scope and forming the iRAT.

Scope

- Working out whether issue is in scope
- Scope defined as "Any software used to enable 'High level Integration' services".
  - For now (‘high level Integration’ is an EOSC term)
- It includes EGI UMD/CMD
  - These are used by services, AND EGI endorses them
- Probably will include the various collaboration services
  - And get experts involved who run such services

iRAT – Issue RAT

- Concept of the iRAT – or Issue Risk Assessment Team is the biggest change in the SVG evolution
- After appropriate experts have been contacted the iRAT is formed which
  - Investigates the vulnerability, and the effect of the vulnerability on the various services
  - Assesses the risk to those services.
  - Works out how to mitigate the problem, whether update software with a patch or carry out other action
  - Drafts appropriate notification/advisory
- Then notification/advisory is sent to the relevant parties defining what actions should be carried out.

Tricky bit

- How to determine scope and form the iRAT?
- For EGI UMD/CMD software – simple to contact the right people
- For the services, the difficult bits are how to find who to contact and what services are using what software
- Ideally a database of software used and contacts/experts for all services in EOSC-hub service catalogue
  - But we don't have it
- To start, we could consider service contacts/security contacts from definitive list of services
  - Except that's not there yet either
- Hoping to start with access to a definitive service list, which EOSC-hub is developing

Therefore

- A long way to go!

Thank you
for your attention!

Questions?

EOSC-hub

eosc-hub.eu @EOSC_eu