EGI-InSPIRE

GridPP
UK Computing for Particle Physics

The EGI Software Vulnerability Group (SVG)

What is a Software Vulnerability?

Most people are familiar with the need to keep their computer systems up to date, whether installing Windows or Linux updates to ensure their systems do not contain known vulnerabilities.

A vulnerability can be seen as a security system that can easily be bypassed, or analogous to a lock that can easily be picked. We define a vulnerability as **"a problem where a principal (e.g. a user) can gain access to or influence a system beyond their intended rights"** This could be where an unauthorized user may gain access to a system. It could be that an authorized user may gain root or administrator privilege, can damage a system, access confidential information, destroy another user's data or impersonate another user.

What is the EGI Software Vulnerability Group?

The purpose of the European Grid Infrastructure (EGI) Software Vulnerability Group (SVG) is **"To eliminate existing vulnerabilities from the deployed infrastructure, primarily from the Grid Middleware, prevent the introduction of new ones and prevent security incidents"**

This is carried out in 3 main ways:

- Handling potential vulnerability problems reported
- Checking software for vulnerabilities (Vulnerability Assessment)
- Educating developers to write secure code

Previously, similar work was carried out by the EGEE Grid Security Vulnerability Group (GSVG)

SVG membership and interaction with other groups

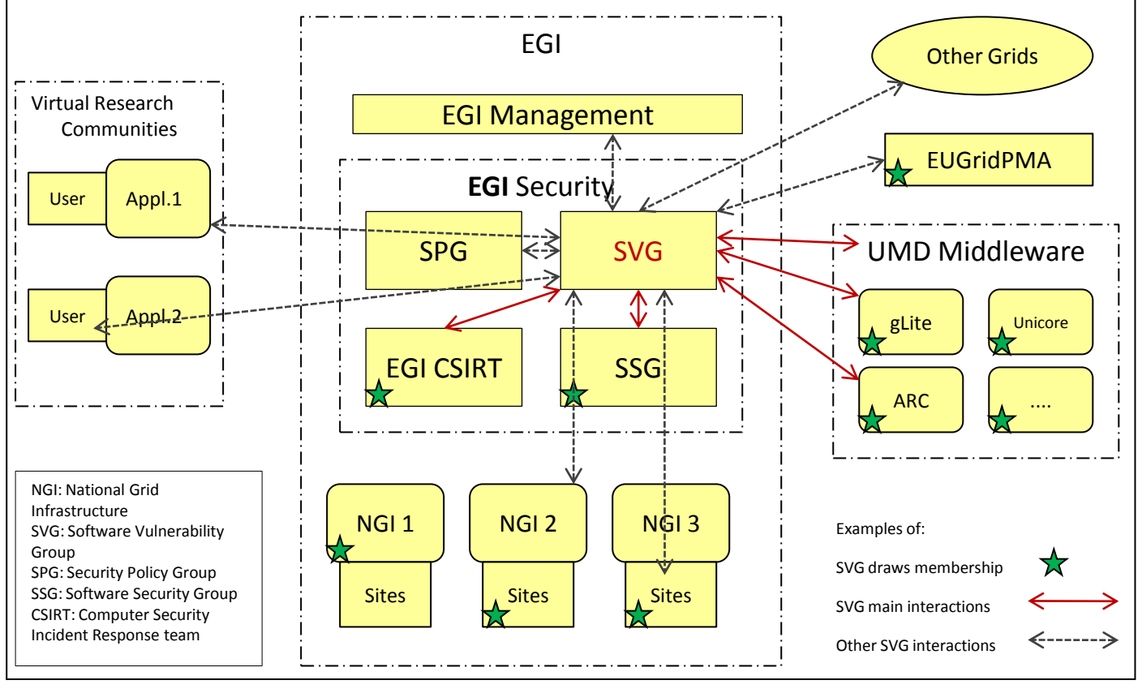
The EGI SVG draws its membership from

Software Providers who contribute the grid middleware distributed and used in the deployed infrastructure

EGI CSIRT Team The Computer Security Incident Response Team who strive to ensure that the deployment is as secure as possible

Sites and NGIs Experienced site administrators and deployment experts who know about security

Others in the grid world with appropriate skills



What if you find a vulnerability?

DO NOT discuss on a mailing list - especially one with an open subscription policy or public archive
DO NOT post information on a web page
DO NOT publicise in any way, e.g. to the media

IMMEDIATELY report it to report-vulnerability@egi.eu

Talking about it in the open will disclose information that will be useful to an attacker

The issue handling process

Most of the Issue handling is carried out by the SVG "Risk Assessment Team" or "RAT"

Investigation

The RAT, along with the reporter and software developers, establish whether the issue is real and what the potential effects of an exploit would be

Risk Assessment

A risk assessment is carried out by the RAT for all valid issues, where the issue is placed in 1 of 4 risk categories – Critical, High, Moderate, Low

Target date for resolution set

This is a fixed value for each risk category

Critical – 3 days
High – 6 weeks
Moderate – 4 months
Low – 1 year

This allows the prioritization of fixing of issues, according to how serious they are. It is then up to the developers and software distributers to ensure the vulnerability is eliminated from the software available to the EGI infrastructure in time for the Target Date

Advisory issued

When the vulnerability is eliminated or on the target date – whichever is the sooner

What is Vulnerability Assessment?

This is the proactive examination of software in order to find vulnerabilities that may exist Various packages have been assessed, significant vulnerabilities have been found and developers been helped with remediation strategies. Assessing further packages is planned. Assessment may be considered prior to allowing new software to be deployed on the EGI infrastructure to help minimize the introduction of new vulnerabilities

Writing secure code

If you are a developer, please make an effort to become aware of how to avoid introducing new vulnerabilities and how to write secure code

Validate input

Don't trust user input, it could be malicious

Check file permissions

Any file or directory with world write permission could be modified with malicious content

Learn about secure programming

Tutorials have been given at various Grid conferences (see further info below), and texts are available on how to avoid writing vulnerable code.

Further information

Vulnerability Assessment information and Secure coding tutorials by Barton Miller and James Kupsch from the University of Wisconsin and Elisa Heymann of the Universitat Autònoma de Barcelona.are available at http://www.cs.wisc.edu/mist/ Much of the work on vulnerability assessment has been carried out by these groups.

Code reviews carried out by Gerard Frankowski and other members of the Security team at the Poznan Supercomputing and Networking Centre in Poland.

SVG Wiki
https://wiki.egi.eu/wiki/SVG

Article in ISGTW at
http://www.isgtw.org/?pid=1002400

SVG RAT members:
Linda Cornwall (RAL), Krzysztof Benedyczak(UWAR), Stephen Burke(RAL), Vincenzo Ciaschini(INFN), Sven Gabriel(Nikhef), Oscar Koeroo (Nikhef), Daniel Kouril (CESNET), Maarten Litmaath (CERN), Mingchao Ma (RAL), Leif Nixon(NSC), Eygene Ryabinkin(RRC-KI), Mischa Sallé (Nikhef), Åke Sandgren(HP2CN), Bernd Schuller(JUELICH), Steve Traylen(CERN), Anders Waananen (UCPH) .

CERN, Switzerland
CESNET, Prague, Czech Republic
HP2CN, Umea University, Sweden
INFN, Istituto Nazionale di Fisica Nucleare, Italy
JUELICH Supercomputing Centre, Germany
Nikhef, The Dutch National Institute for Subatomic Physics, Amsterdam, The Netherlands
NSC, Linkoping university, Sweden
RAL, The Rutherford Appleton Laboratory, UK
RRC-KI, The Russian Research Centre "Kurchatov Institute", Moscow, Russia
UCPH, Niels Bohr Institute, Copenhagen, Denmark
UWAR, Poland

e-infrastructure

http://www.egi.eu/policy/groups/Software_Vulnerability_Group_SVG.html