# EGI.eu

# THE SOFTWARE VULNERABILITY ISSUE HANDLING PROCESS

| | |
|---|---|
| Document identifier | EGI-SVG-VulnerabilityHandling-717-V2.doc |
| Document Link | https://documents.egi.eu/document/717 |
| Version | V2 |
| Policy Group Name | EGI SVG |
| Contact Person | Dr Linda Cornwall/STFC |
| Document Type | Procedure |
| Document Status | FINAL |
| Approved Date | 24/10/2011 |


Abstract

In order to reduce the risk of computer security incidents, it is important to handle and resolve software vulnerabilities reported in the EGI infrastructure. This document describes the process for Grid Software Vulnerability Issue handling by the EGI InSPIRE project. It describes what is meant by vulnerability, how to report a vulnerability, and how vulnerabilities are handled. It describes the responsibilities of various people within the Software Vulnerability Group (SVG), the EGI InSPIRE project and in the communities providing software distributed in the EGI Unified Middleware Distribution and how the various groups interact with this process.

## I. COPYRIGHT NOTICE



## II. AUTHORS LIST

| | Name | Partner/Activity/ Organisation/Function | Date |
|---|---|---|---|
| **From** | Dr Linda Cornwall | STFC | |

## III. DOCUMENT LOG

| Issue | Date | Comment | Author/Partner |
|---|---|---|---|
| 0.1 | 19/07/2011 | Revised version after 1 year | STFC/Dr Linda Cornwall |
| 0.2 | 22/07/2011 | Changes from comments – mainly more clarity about different types of software | STFC/Dr Linda Cornwall |
| 1.0 | 27/07/2011 | Changes to separate from Milestone MS405 | STFC/Dr Linda Cornwall |
| 2.0 | 13/09/2011 | Address Michel Dresher comments | STFC/Dr Linda Cornwall |
| 3.0 | 28/09/2011 | Change for Non UMD software (CSIRT subgroup) | STFC/Dr Linda Cornwall |
| 4.0 | 11/10/2011 | Changes from Tiziana Ferrari's comments | STFC/Dr Linda Cornwall |
| V2 | 24/10/2011 | Renamed to move to location of approved version | STFC/Dr Linda Cornwall |

## IV. APPLICATION AREA

This document is a formal EGI.eu policy or procedure applicable to all participants and associate participants, beneficiaries and Joint Research Unit members, as well as its collaborating projects.

## V. POLICY/PROCEDURE AMENDMENT PROCEDURE

Reviews and amendments should be done in accordance with the EGI.eu "Policy Development Process" (https://documents.egi.eu/document/169).




## VI. ORGANISATION SUMMARY

To support science and innovation, a lasting operational model for e-Infrastructure is needed − both for coordinating the infrastructure and for delivering integrated services that cross national borders. The objective of EGI.eu (a foundation established under Dutch law) is to create and maintain a pan-European Grid Infrastructure in collaboration with National Grid Initiatives (NGIs) in order to guarantee the long-term availability of a generic e-infrastructure for all European research communities and their international collaborators.

In its role of coordinating grid activities between European NGIs, EGI.eu will:

- Operate a secure integrated production grid infrastructure that seamlessly federates resources from providers around Europe
- Coordinate the support of the research communities using the European infrastructure coordinated by EGI.eu
- Work with software providers within Europe and worldwide to provide high-quality innovative software solutions that deliver the capability required by our user communities
- Ensure the development of EGI.eu through the coordination and participation in collaborative research projects that bring innovation to European Distributed Computing Infrastructures (DCIs)

The EGI.eu is supporting 'grids' of high-performance computing (HPC) and high-throughput computing (HTC) resources. EGI.eu will also be ideally placed to integrate new Distributed Computing Infrastructures (DCIs) such as clouds, supercomputing networks and desktop grids, to benefit the user communities within the European Research Area.

EGI will collect user requirements and provide support for the current and emerging user communities. Support will also be given to the current heavy users of the infrastructure, such as high energy physics, computational chemistry and life sciences, as they move their critical services and tools from a centralised support model to one driven by their own individual communities.

The EGI community is a federation of independent national and community resource providers, whose resources support specific research communities and international collaborators both within Europe and worldwide. EGI.eu, coordinator of EGI, brings together partner institutions established within the community to provide a set of essential human and technical services that enable secure integrated access to distributed resources on behalf of the community.

The production infrastructure supports Virtual Research Communities − structured international user communities − that are grouped into specific research domains. VRCs are formally represented within EGI at both a technical and strategic level.

**Glossary**

| | |
|---|---|
| CSIRT | (The EGI) Computer Security Incident Response Team |
| CSIRT sub-group | Sub-group of CSIRT who are involved in vulnerabilities in non-UMD software and potentially critical vulnerabilities |
| DMSU | (The EGI) Deployed Middleware Support Unit |
| EGEE | The EU Enabling Grids for E-science project |
| GSVG | (The EGEE) Grid Security Vulnerability Group |
| IRTF | The (EGI) Incident Response Task Force |
| NGI | National Grid Infrastructure |
| RAT | The Risk Assessment Team |
| SVG | (The EGI) Software Vulnerability Group |
| TD | Target Date |
| UMD | (The EGI) Unified Middleware Distribution |

## VII. EXECUTIVE SUMMARY

To ensure the sustainability of the deployed EGI infrastructure it is important that it is sufficiently secure. The main purpose of security is to allow people to enjoy the benefits they are entitled to. If the infrastructure is not secure, for example if users' data is destroyed or exposed, or users cannot use the system because it has been damaged then users will demand other means of carrying out their activities. If incidents happen where sites contributing to the EGI infrastructure are compromised due then sites will not wish to participate and provide resources to the grid.

A large part of ensuring that the infrastructure is secure is to ensure that the software deployed is secure, by eliminating existing software vulnerabilities and preventing the introduction of new ones. This is the task of the EGI Software Vulnerability Group. This document describes how software vulnerabilities found or reported are handled.

The basic process is that:

- Anyone may report a vulnerability by e-mail to report-vulnerability@egi.eu
- The Risk Assessment Team, along with the reporter and developer investigate the issue, to see if it is valid.
- If a reported issue is found to be valid, the Risk Assessment Team place the issue in one of four risk categories – Critical, High, Moderate, or Low.
- According to the risk category, a fixed target date for fixing this vulnerability is set.
- The developers then should try and fix the issue by the target date.
- An advisory is released when a fixed version of the software is released, or on the target date, whichever is the sooner.

This document describes this process in more detail, and defines the responsibilities from the point of view of the various parties involved, i.e. the Reporter of the Issue, The Software Vulnerability Group, The Software providers, the EGI Deployed Middleware Support Unit, Security Operations and the sites.

The full process applies to software distributed as part of the EGI Unified Middleware Distribution (UMD) from providers with which EGI has a Service Level Agreement (SLA). The relevance and Risk in the EGI infrastructure of Vulnerabilities in other software widely deployed on the EGI infrastructure is also assessed.

This is an updated version with some minor changes after 1 year's operation of the procedure, and since changes have been approved it replaces the version produced as milestone MS405 [R 3] and is moved to the permanent location of this procedure.

# TABLE OF CONTENTS

# 1 INTRODUCTION

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. Vulnerabilities also may occur in the Grid Middleware and other software used in the EGI Grid Infrastructure. The purpose of the EGI Software Vulnerabilities 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." The purpose of this document is to describe the EGI Software Vulnerabilities Group (SVG) and the process for handling software vulnerabilities found in the EGI infrastructure, with the main focus being on Vulnerabilities found in the Grid Middleware, which is supplied by EGI as part of the EGI Unified Middleware Distribution (UMD), and additionally vulnerabilities found in operational tools developed by the EGI InSPIRE project.

This is updated after 1 year's use of the procedure, as approved as part of MS405, [R 3], as a result of experience and clarification of some information.

# 2 SCOPE OF THE SVG ISSUE HANDLING ACTIVITY

## 2.1 Background

In 2005 it was recognised that something should be done to log and try to fix/eliminate vulnerabilities in the grid infrastructure in order to reduce the likelihood of incidents. The Grid Security Vulnerability Group (GSVG) was included in the EGEE-II proposal. A process for handling vulnerabilities, setting a target date for fixing the issue, and responsible disclosure was established and approved by the EGEE management for software produced by EGEE [R 1]. Such an activity is also recognized as being necessary for software used in the EGI infrastructure, and the EGI Software Vulnerabilities Group (SVG) was included in the EGI proposal.

## 2.2 Changed Situation for EGI

For the EGEE GSVG, the focus was very much on software provided as part of gLite. Permissions for releasing information on vulnerabilities according to the agreed process were only admissible for gLite, as it was part of the EGEE project. For EGI, the situation is different. The EGI InSPIRE project is distributing software provided by 3rd parties, by the EGI Middleware Unit as part of the EGI Unified Middleware Distribution (UMD). A service level agreement between EGI and the software providers is being defined which gives the SVG permission to handle vulnerabilities according to the agreed process, and which has the software providers agree to a response time if a potential vulnerability is found in the software they supply.

Some other changes to the process are being made. The amount of time allowed to fix vulnerabilities after the assessments are complete has been lengthened, partly to allow a more realistic schedule for all but the most critical issues, and partly because in EGEE the timescales were probably set much stricter than needed.

## 2.3 Scope of SVG

Between EGI Computer Security Incident Response Team (CSIRT) and EGI SVG all problems concerning security of the deployed EGI infrastructure should be dealt with. It is worth noting that sites are responsible for their own security, CSIRT will advise and recommend on security matters and have the power to suspend sites from the infrastructure if they fail to apply critical security patches.

The handling of incidents is the responsibility of the EGI Incident Response Task Force (IRTF) and they are handled according to the Incident Response Procedure [R2]. However, if an incident turns out to be due to a software vulnerability then the SVG may get involved. SVG should ensure that the software available for installation on the EGI infrastructure is sufficiently secure and contains as few vulnerabilities as possible, thus reducing the likelihood of incidents.

The main task of SVG is to handle vulnerabilities reported in the software distributed by EGI (in the UMD) in the manner described in this document and defined in section 2.5. Such vulnerabilities are generally not handled elsewhere hence the SVG provides the mechanism for handling such vulnerabilities. For most of this software EGI has a Service Level Agreement with the software providers, who agree that vulnerabilities in this software is handled by the EGI SVG and agree to appropriate response times. Additionally, SVG handles vulnerabilities in Operational tools developed by the EGI InSPIRE project which are not in the UMD.

Vulnerabilities in 3rd party software distributed in the UMD including dependencies for which there is a patch from the provider may also need to be handled by SVG. If an issue is reported to SVG and the provider is not aware of it then the SVG will forward information to the software provider. If the provider announces a patch (the more likely case) then SVG may need to produce a risk assessment.

The risk posed by the vulnerability in the EGI infrastructure needs to be assessed by SVG to establish the timescale for which the EGI UMD should include the fixed dependency.

For other software used in the EGI Infrastructure the SVG will, jointly with CSIRT consider the risk in the EGI infrastructure. In this case, an agreed subgroup of CSIRT will be added to the issue. If it is reported to SVG rather than to the software provider the SVG will also pass information onto the software provider.

The SVG will not handle vulnerabilities in general software which is not normally installed on the EGI infrastructure.

SVG will take some action on any vulnerability reported, for example if a vulnerability is discovered and reported that is completely out of scope: SVG will at least attempt to forward that information to the software provider if it is not clear that they are already aware of it. The reporter will also be told what action is taken, including if no action is taken.

| Software Source | S/W provider aware/announced vulnerability | S/W provider not clearly aware of vulnerability | Risk Assessment | Other comment |
|---|---|---|---|---|
| **EGI UMD – e.g. EMI/IGE software for which EGI has SLA** | Problem fully handled according to process in this document by SVG | | SVG | |
| **Operational Tools developed by the EGI InSPIRE project** | Problem fully handled according to the process in this document by SVG, except distribution of tools not in UMD | | SVG | |
| **Linux Operating system software on which the EGI infrastructure is based** | CSIRT sub-group /SVG investigates relevance to EGI | Inform software provider | SVG/CSIRT subgroup jointly | Usually CSIRT member will contact provider if necessary |
| **EPEL software (Extra Packages for Linux Enterprise)** | CSIRT sub-group /SVG investigates relevance to EGI | Inform software provider | SVG/CSIRT subgroup jointly | SVG or CSIRT member will contact provider depending on knowledge |
| **Other Software widely installed on the EGI Infrastructure** | CSIRT sub-group /SVG investigates relevance to EGI | Inform software provider | SVG/CSIRT subgroup jointly | SVG or CSIRT member will contact provider depending on knowledge |
| **Software not installed on the EGI infrastructure** | Do nothing | Inform software provider | None | Only action is to forward information. |

This table is a guide to how various cases are handled. Who carries out actions for software that is not part of the EGI UMD for which EGI has an SLA largely depends on knowledge and experience and whether anyone has any past contacts. A new subgroup, comprising mainly of CSIRT members but not including the whole CSIRT Team has been formed and these people jointly with the SVG RAT

members assess issues which do not concern EGI UMD middleware. This is known as the CSIRT sub-group.

In summary, the main purpose of SVG is to handle vulnerabilities in software distributed as part of the EGI UMD according to this document. However, for other software widely deployed on the Grid Infrastructure the relevance and if appropriate the risk and what action (if any) to take will be assessed jointly by the SVG and the CSIRT sub-group.

## 2.4 Scope of this Document

This document describes how specific potential vulnerability issues reported to or found by the EGI Software Vulnerability Group are handled. It describes the interfaces between the various groups involved in handling issues. It does not cover other activities, such as checking code and assessing software for vulnerabilities, ensuring new code introduced into the EGI infrastructure is secure, or developer education. It does include the handling of vulnerabilities found as a result of assessing software for security, which are handled in the same way as vulnerabilities found or reported in other ways.

## 2.5 What is a vulnerability

There are many definitions of a software vulnerability. We usually consider a vulnerability as a problem where a principal 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. This could be where a user gains privileges they should not be able to hold, such as root or administrator privilege, can damage a system, gain access to data or information that is confidential, or impersonate another user. It can also be if a user is able to cause damage to a 3rd party via usage of the system.

Some people who carry out vulnerability assessments do not report issues if they cannot develop an exploit. SVG does require a proof of concept piece of software to be developed in order for a problem to be treated as vulnerability. Dangerous coding constructs, where there is a possibility that an exploit can be developed, can be considered to be vulnerabilities. However, if the risk is considered to be negligible then the issue may be treated in another way, e.g. as a bug, as the people assessing the issue considers appropriate.

## 2.6 What is NOT a vulnerability

### 2.6.1 Actions that can only be carried out by site administrators

In general, site administrators are (almost) trusted at the sites they manage – and they are assumed to be able to access and manipulate data stored on their equipment. The only thing that they are not trusted with is bulk encrypted data and encryption keys. Site administrators should not be able to decrypt encrypted data at will, however as data needs to be decrypted for processing it cannot be entirely protected from processes with site administrator privileges.

### 2.6.2 Issues which provide information that may be useful to an attacker

If information is provided which may be of use to an attacker, but does not represent an exploit in itself, this is not necessarily considered to be a vulnerability. In the past such issues have been treated as ‘Low’ risk issues, even if there is virtually no risk. These can again be rejected, treated as standard bugs or as vulnerabilities as the RAT considers appropriate.

### 2.6.3 General Concerns

This is the type of report where someone states that ‘this may not get installed correctly’ or ‘some users will do this incorrectly’. Such concerns will not be considered vulnerabilities, but can be raised with the appropriate groups. If they are reported to SVG then SVG will raise them to the appropriate groups.

# 3 ISSUE HANDLING PROCESS

## 3.1 The Risk Assessment Team (RAT)

The Risk Assessment Team (RAT) is the group of people within the Software Vulnerability Group (SVG) who carry out the issue handling process of the SVG, and are party to information on vulnerabilities which have not been disclosed publically. As the phrase Risk Assessment Team implies, one of their main duties is to assess the risk associated with a software vulnerability found, so that a software vulnerability can be fixed in a timely manner according to the severity of the problem.

The RAT members include developers from the various software provider teams whose software is included in the EGI UMD, CSIRT members, NGIs and experienced site administrators.

Some members of the RAT (in particular the chair of the activity) also co-ordinate the activity to ensure that the process is carried out as stated in this document. These are members of the EGI community. This includes making sure that contact details for the developers are in available the infrastructure is in place, and the various parts of the process are carried out in a timely manner.

## 3.2 Basic process

### 3.2.1 Reporting an issue

If anyone finds a suspected vulnerability, they should report it to

report-vulnerability@egi.eu

It is then entered into the Software Vulnerability Issue Tracker. (This happens automatically if the vulnerability is reported to this e-mail address.) This is a private tracker, information can only be accessed by the RAT and others involved in the fixing of the issue.

### 3.2.2 Investigation of issue

The RAT, in conjunction with the reporter of the issue and the developers of the affected software, investigate the issue. This is in order to establish whether or not there is an issue, and if there is what the problem is, in what circumstances it may be exploited and what the probable effect of exploitation is.

If as a result of this investigation it is agreed that no problem exists then no further action is taken.

### 3.2.3 Risk Assessment

Assuming there is a real problem a risk assessment is carried out by the RAT. The RAT discusses the impact of each issue in the EGI Grid environment. The RAT then places the issue in one of 4 risk categories:

- Critical
- High
- Moderate
- Low

The category is established by vote, i.e. the RAT members vote in which risk category an issue should be placed. Usually a clear majority of the votes are for a particular risk category, which is then taken as the resulting risk category. When the votes remain divided after ample discussion, either the more conservative (i.e. higher) level is taken, or the matter may end up deemed out of scope for the SVG, as explained in section 2.6. In some such cases the CSIRT may be asked to consider an operational

advisory. In others it may be concluded that there is no vulnerability, and the RAT and the software provider may consider whether other action needs to be carried out, such as submission of a standard bug.

### 3.2.4 Target Date Set

A Target Date (TD) for fixing is set according to the risk category, as below.

- Critical – 3 days (see section 3.3)
- High – 6 weeks
- Moderate – 4 months
- Low – 1 year

This is from the day that the risk category is set. The reason for this is to allow the prioritization according to severity and timely fixing of vulnerabilities in the software.

The software providers, UMD, and reporter are informed of the risk category, and of the target date for fixing the issue.

The SVG aims to reach this point, i.e. where the risk category is set, within at most 4 working days of an issue being reported. Usually this should be done within 2-4 working days, which is a realistic aim to allow time to contact the RAT and the developers, investigate the problem, find the likely impact if the issue were to be exploited and assess the risk. This may be done more quickly in the case of issues assessed as critical, see section 3.3

### 3.2.5 Fixing the issue

It is then between the software providers, packagers and the EGI Deployed Middleware Support Unit (DMSU) to try and ensure that the issue is fixed by the target date. All the appropriate parties are contacted by SVG by e-mail, told of the outcome of the risk assessment and the target date, and added to the issue in the Software Vulnerability issue tracker. SVG does not co-ordinate the fixing and release of the software. The EGI DMSU should ensure that the version of the software available in the UMD on the target date for installation across the EGI infrastructure does not contain the vulnerability.

### 3.2.6 When the issue is resolved

When the new version of the software is released with the issue resolved an advisory is issued by the SVG. For software distributed as part of the EGI UMD, this is when the software is available in the UMD. The release notes for any software distributed as part of the UMD should refer to the advisory, and the advisory should refer to the release notes.

### 3.2.7 If the target date is reached and no patch is available

The advisory is released on the target date, or the first working day after the target date. This may not necessarily be the case for Critical issues, see section 3.3.

## *3.3 Special process for critical risk issues*

It is usually apparent quite quickly if an issue falls into one of the higher risk categories, and investigation tends to happen quickly. Hence in this case the aim is to investigate the issue and assess the risk within one working day. It is probably more important to simply establish whether the problem is real and find a short-term solution, than carry out a full investigation and decide on a long-term solution.

While it is hoped that these will be rare, it should be noted that if a critical issue does occur a special process will be carried out. What should be done will need to be considered on a case-by-case basis,

and this is carried out jointly with the CSIRT sub-group. This provides a guide for what might be done.

### 3.3.1 Alert all appropriate parties

SVG will alert the CSIRT sub-group, developers, and the EGI DMSU as appropriate as soon as a critical vulnerability or potentially critical vulnerability has been identified. Alert is usually by e-mail. Appropriate parties are also added to the issue in the Software Vulnerability Issue tracker.

### 3.3.2 Consider sending a 'heads up'

This is an alert to sites that a serious problem has been found and that further advice will follow. Whether this is appropriate is mainly the decision of the CSIRT sub-group, but also with input from SVG if appropriate.

### 3.3.3 Establish the effect of the exploit in the EGI infrastructure

Make sure that the effect of the exploit in the EGI infrastructure is as clearly established as possible.

### 3.3.4 Establish in what situation the vulnerability can be exploited

Establish what software or combination of software/operational configuration allows the vulnerability to be exploited.

### 3.3.5 Find how widespread the problem is likely to be

If the problem only occurs on a few sites, then it may be appropriate to ask those sites to change something, rather than handle as a widespread critical vulnerability. If the problem affects a large proportion of EGI sites, then there is a large-scale problem.

### 3.3.6 Find out how quickly a patch can be made available

Find out whether it is possible to produce a patch in around 3 days, and if not how quickly a patch can be made available.

### 3.3.7 Decide whether to wait for a patch

If a patch can be made available in 3 working days, normally no action will be recommended. If a patch will take longer, then the decision needs to be made alongside 3.3.8 as to whether to wait for a patch or recommend other action.

### 3.3.8 Find if other action can mitigate or resolve the problem

If a patch cannot be made available quickly, the CSIRT sub-group and SVG along with the developers may be able to come up with some mitigating action. A small configuration change may be sufficient to prevent the vulnerability being exploited. If this is the case, establish what needs to be done in what circumstances. Test any changes that are recommended. Care should obviously be taken not to recommend changes in a hurry that do not work, worsen the situation, or inadvertently prevents a site from operating.

### 3.3.9 Carry out any interim action

If interim action is recommended, produce the appropriate advisory. Otherwise, if not already issued, a heads up as in 3.3.2 may be issued. The decision may be to do nothing. Usually the advisory for interim action is not released publicly, but sent to appropriate lists.

### 3.3.10 Ensure advisory is completed ready for the software release

Make sure the advisory is drafted and agreed with the software provider ready for the release of the software.

### 3.3.11 EGI CSIRT Handles Critical vulnerability

After this, it is the job of EGI CSIRT to handle the critical vulnerability according to the EGI CSIRT Critical Vulnerability Operational Procedure [R 4].

## *3.4 Issuing advisories*

Advisories are normally issued publicly on the EGI Wiki at
https://wiki.egi.eu/wiki/SVG:Advisories

The EGI CSIRT Team, sites, along with the reporter of the issue will then be informed of the availability of the new advisory. The following lists, along with the original reporter of the problem, will receive the advisory.
Egi-csirt-team@mailman.egi.eu
NGI-Security-contacts@mailman.egi.eu
Site-security-contacts@mailman.egi.eu
NOC-managers@mailman.egi.eu

Advisories should include the type of problem that would occur if the vulnerability were to be exploited, but not include how to exploit the vulnerability.

## *3.5 Principles of dealing with other situations*

While the majority of issues are expected to result from bugs in the software included in the UMD, a minority of issues are likely to fall into the outlined categories below.

### 3.5.1 Operational Vulnerabilities

Some issues may turn out to be purely operational, and no software fix is required. In this case CSIRT is informed of the problem with any appropriate recommendations.

### 3.5.2 Issues where the decision is not to fix

This may be because there isn't a practical way of fixing it, or the problem is part of the design of the system. In this case CSIRT will be informed, with recommendation of any mitigating action that should be taken or problems they should be alert to.

### 3.5.3 Issues concerning other software

For vulnerabilities reported concerning software other than middleware distributed in the UMD, but used in the EGI infrastructure SVG will inform the CSIRT sub-group. SVG and the CSIRT sub-group will jointly consider the risk. If the software provider has not been informed the information will be forwarded to the software provider. For issues concerning software that is not installed on the EGI

infrastructure the SVG will forward to the software provider if it is not clear whether the software provider knows about it, but otherwise do nothing. SVG will always inform the reporter (usually by e-mail) of what action is or is not taken and why.

### 3.5.4 Other cases

The principle is that any issue where there is an exploitable vulnerability should be dealt with in some way, but not in a way that provides information publicly that is useful to a potential attacker.

# 4 REPORTERS VIEW AND RESPONSIBILITIES

## 4.1 Not publicising a vulnerability

It is important that information on vulnerabilities is kept private while they are investigated and while the software providers are fixing them. Vulnerabilities must not be entered on any publicly readable bug tracking system, discussed on any mailing list that is either publicly archived or does not have a strictly controlled membership policy, or placed on any web page.

Vulnerabilities should not be publicised in any way without agreement from the SVG.

If a vulnerability has been distributed publicly, e.g. on a less secure mailing list, or on a publicly accessible web page, then the reporter should make this known to the SVG and if possible try to ensure the information is removed.

## 4.2 Reporting a vulnerability

Anyone who finds a vulnerability should report it to the EGI SVG via report-vulnerability@egi.eu

## 4.3 Help and co-operate with the investigation

While this is not mandatory, it is can be extremely helpful if the person who finds a vulnerability is able to assist with the investigation.

## 4.4 Reporter receives information

The reporter will receive information on the outcome and conclusion of the investigation, including the risk category and Target Date, and will receive a copy of the advisory.

# 5 SOFTWARE VULNERABILITY GROUP (SVG) VIEW AND RESPONSIBILITIES

## 5.1 Set up and maintain infrastructure for issue handling

The Software Vulnerability Group (SVG) will set up and provide the infrastructure for issue handling. This includes the mailing list for reporting, the Software Vulnerability Issue tracker, the mailing list for the RAT to investigate and assess issues, and the web pages for release of advisories. It also involves ensuring that the contact details for the various software providers are at hand and readily available.

## 5.2 Provide a rota for cover on working days

SVG will try and ensure that a person is available to respond to any issue reported and carry out the issue handling process on all working days. This will be known as the SVG duty. The chair of the SVG will organise this rota. The majority of the time the chair will be on duty, if not available one the deputies will take duty, if none are available another RAT member may be able to take on a duty.

Note that SVG does not guarantee cover on all working days, but will aim to do so. SVG does not guarantee out of hours support, or cover over public holidays – but most members do check their e-mails and will deal with any serious or urgent problems on a best effort basis.

## 5.3 When a potential issue is reported

Anyone may report an issue – by e-mailing report-vulnerability at egi.eu

The SVG duty should do the following:

- Acknowledge the reporter.
- Contact the provider of the software (unless the issue is quickly deemed invalid, or the reporter is informing the SVG of a vulnerability reported and fixed by e.g. an operating system provider).
- Ensure that the issue is in the Software Vulnerability Issue tracker, (if it has not been reported via the report-vulnerability e-mail).
- Alert the Risk Assessment Team (RAT) that a new issue has been reported by e-mail including “RAT alert” in the title.

This should happen as soon as possible, typically within an hour or two, or at least within 1 working day.

## 5.4 If information has been made public

Although we ask people to take care when discussing vulnerabilities it is important to consider the case where information may have accidentally or intentionally been made public. If this happens, then we should consider a special process for dealing with this. If possible, such as if it is on a web page managed by people contributing to or related to EGI, the information should be removed. When assessing the risk, if information is public it may be that the issue is placed in a higher risk category than it would be if the RAT were confident of its privacy. In most cases, CSIRT will be informed that the vulnerability has been disclosed and in what way it has been disclosed.

## 5.5 Investigation of an issue

The SVG RAT along with the reporter and the provider of the software investigate whether or not there is a vulnerability. It should also be established what the effect of an exploit might be, and in what circumstances the vulnerability may be exploited.

If there is not a problem at all, the issue is closed. If the issue needs attention but is not a software vulnerability, then appropriate action is taken as described e.g. in section 3.5.

## 5.6 Risk Assessment

If the issue is valid a Risk Assessment is carried out by the RAT which discusses the impact of each issue in the Grid environment. The SVG duty will call for a Risk Assessment. For each valid issue, the Risk Assessment Team places the issue in one of 4 Risk Categories

- Critical
- High
- Moderate
- Low

The category is established by vote, i.e. the RAT members vote in which risk category an issue should be placed. Usually a clear majority of the votes are for a particular risk category, or a consensus is reached, which is then taken as the resulting risk category. When the votes remain divided after ample discussion, in some cases the more conservative (i.e. higher) level is taken, or the matter may end up deemed out of scope for the SVG as explained in section 2.3. In some cases the CSIRT may be asked to consider an operational advisory.

The Risk Assessment should be discussed on the RAT list, not in the tracker entry, and a summary placed in the tracker entry.

## 5.7 Target Date Set

When the Risk has been established, the SVG on duty sets the Target Date (TD) for fixing, according to the risk category, as below.

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

This is from the day that the risk category is set. The reason for this is to allow the prioritization according to severity and timely fixing of vulnerabilities in the software. The SVG duty will then:

- Set the risk and TD in the tracker
- Alert the software supplier, EGI DMSU, and reporter of the risk category and the TD. This should be done according to agreed contact details established with the various parties
- Add the contacts to the tracker, so they can view information on the vulnerability and the conclusions of the risk assessment.

The SVG aims to reach this point, i.e. where the risk category is set, within at most 4 working days, of an issue being reported. For critical risk issues, the aim is to reach this point within 1 working day if possible (see section 3.3).

## 5.8 Provide help and advice where needed on how to resolve an issue

It is not an SVG task to fix vulnerabilities. Some members of the SVG RAT are drawn from development teams, so they may happen to be involved. However the SVG RAT will provide help and advice on how to fix or mitigate problems whenever possible.

## 5.9 Draft Advisory

The SVG duty produces a draft of the advisory, with input from RAT members, the software provider and reporter as appropriate. The contents should be agreed with the software provider and the Risk Assessment Team.

The SVG duty puts a placeholder file on the web page – containing no information except to state that this has not been released yet.

## 5.10 When the software is released/or on the target date

The SVG duty makes any modification necessary to the advisory, e.g. to refer to the release notes, states the date of release and uploads it to the web page.

The following lists, along with the original reporter of the problem, will receive the advisory.

Egi-csirt-team@mailman.egi.eu

NGI-Security-contacts@mailman.egi.eu

Site-security-contacts@mailman.egi.eu

NOC-managers@mailman.egi.eu

The advisory will also normally be placed on the EGI Wiki at

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

# 6 SOFTWARE PROVIDERS VIEW AND RESPONSIBILITIES

## 6.1 Software providers agreed to an SLA

By having their software in the UMD, software providers should have agreed to a Service Level Agreement (SLA) with includes agreeing:

- That any suspected vulnerabilities found in their software are handled using the EGI SVG issue handling process.
- To provide contact details for their development teams.
- To respond when asked by the SVG as soon as possible – or at least within 2 working days.
- To co-operate to ensure that if they find a vulnerability in their own software, they fix it in a timely manner and ensure that the new version is available in the UMD for an appropriate amount of time prior to releasing information on this.

A revised version of this process document will be made available when the SLA is in place referring to the SLA.

## 6.2 Software providers supply up to date contact details

Software providers should ensure that they provide up to date contact details so they can be contacted as soon as possible in the event of a potential vulnerability being reported for their code.

It is recommended that software providers supply e-mail addresses both of development teams and of an overall responsible person. This should ensure that developers can be involved in the investigation whenever possible, and the overall responsible person for the software suite is alerted to any potential problems when they occur.

## 6.3 Software providers co-operate with the investigation

The providers of the software will be alerted with an e-mail with a title including
"SVG Alert – Possible Vulnerability in software".
Software providers should:

- Respond as soon as they see the e-mail.
- Respond anyway within 2 working days, preferably 1 working day for issues deemed Critical.
- Help with the investigation as to whether the issue is real or not, and in what circumstances it may be exploitable

If the investigation concludes that there is a software vulnerability then:

## 6.4 Await Risk Assessment

The development team may usually wait for the risk category and TD to be set by the RAT.

## 6.5 Ensure a fixed version is available by the Target Date

It is the responsibility of the software providers to try and ensure a version free from the vulnerability is available in the UMD by the target date. The developers may consult the SVG who will help where they can with advice on how to fix the problem, and will need to co-ordinate with the UMD to ensure that the software is released on time. The software provider will also need to make it clear to the UMD when a new version fixes a vulnerability.

## 6.6 Review advisory

The advisory should be agreed between the development team and the RAT.

## 6.7 When software providers find a vulnerability

If a software provider team finds a vulnerability in their own software, they must ensure appropriate action is taken to resolve the vulnerability in a timely manner. They must ensure that a fix can be made available in the UMD prior to disclosing information on this vulnerability. There are two ways this may be achieved:

### 6.7.1 Inform SVG as soon as they find the vulnerability

This is what SVG would strongly prefer. This is important because there is always the possibility that if the developers can find the problem others could, especially as the software provided by the UMD is mostly open source. If this is done, the vulnerability is handled in the same way as other vulnerabilities. This also has the advantage to the development team of being able to ask the SVG for any help and advice needed in resolving it. It also allows the SVG to provide advice to software providers to prevent them accidentally disclosing the problem, which can occur by use of a publicly readable bug tracker or open source software distribution system.

### 6.7.2 Fix the vulnerability prior to informing SVG

Some software providers will inevitably fix the vulnerability prior to informing SVG. If this is done, the software provider should report it to the report-vulnerability at egi.eu list after they have fixed it, explaining the problem and how it has been resolved. The SVG RAT will then carry out a risk assessment and set the target date in the usual way. The RAT and the development team should agree on an advisory, which will be released when the fixed version of the software is available in the UMD. If software providers take this approach they need to be aware that if there were to be an incident whereby such a vulnerability is exploited and they had delayed fixing it, it would be bad for both EGI and their own reputation.

## 6.8 Software providers are invited to join the SVG

Members of the RAT are drawn from sites, NGIs, CSIRT team, and software providers. Software providers are invited to provide a RAT member. The workload induced on a RAT member should only be a small percentage of that person's time. It would be best if the RAT includes members from all the major software suppliers to maximize the knowledge base of the RAT and efficiently investigate and assess problems. One incentive to provide membership is the opportunity to influence the process as well as helping to provide a secure infrastructure.

# 7 EGI DEPLOYED MIDDLEWARE SUPPORT UNIT VIEW AND RESPONSIBILITIES

As the focus of the EGI SVG will be on ensuring that the software released in the UMD is as secure as possible, and the majority of the issue handling work is expected to concern this, the EGI DMSU who will inevitably need to interact with this process. This of course only applies to vulnerabilities in software distributed by the UMD.

## *7.1 The EGI DMSU will be alerted when a Risk Assessment is complete*

The SVG will alert the agreed contacts at EGI DMSU by e-mail when a risk assessment is complete, stating the target date for fixing of the problem, and the software involved. These contacts will also be added to the issue in the Software Vulnerability issue tracker, so they can view the issue.

## *7.2 The EGI DMSU and Software provider work to provide a new version in time on TD*

The EGI DMSU and the software provider will need to co-ordinate their work to ensure that a new version of the software, with the vulnerability fixed, is available in the UMD on or before the target date. In some cases, such as issues categorized as critical or high risk, an emergency release may need to be made available. Note that detailed information will be in the Software Vulnerability issue tracker, and not in any public ticketing or bug tracking system. However, if needed a 'mirror' with little information may be placed in another system, to help DMSU with workflow management.

## *7.3 The EGI DMSU informs SVG when about to release a version which fixes a vulnerability*

The EGI DMSU should inform SVG when they are about to release software which fixes a vulnerability. This allows SVG to complete the advisory as appropriate and refer to the release version.

## *7.4 The EGI DMSU ensures release notes refer to the advisory*

The Release notes should refer to the advisory (just as the advisory refers to the release notes).

# 8 CSIRT TEAM VIEW AND RESPONSIBILITIES

In EGI, the SVG and CSIRT teams work closely together to ensure the security of the EGI infrastructure. Several members of the CSIRT team are in the SVG, so are alerted when a new vulnerability is found.

## 8.1 CSIRT Team may report a vulnerability

CSIRT members may find a security problem that turns out to be due to a vulnerability, in which case they may report it as in section 4.

When the IRTF is handling incidents, if an incident turns out to be due to a software vulnerability in the UMD distribution, they should report it to the SVG. A vulnerability that has caused an incident is likely to be classed as critical or at least high risk.

## 8.2 CSIRT sub-group will be informed if a vulnerability is assessed as critical

If the SVG identifies a vulnerability that is Critical, then CSIRT sub-group is informed. If it is not possible to produce a fixed version of the software on a short timescale, SVG will work with the CSIRT sub-group to decide how best to mitigate the problem.

## 8.3 CSIRT sub-group will be informed if an operating system vulnerability is reported to SVG

If a vulnerability is reported that concerns the operating system, or other non EGI UMD software CSIRT sub-group will be informed. This allows the people with the most knowledge to handle or mitigate the impact to the EGI.

## 8.4 CSIRT Team may issue an operational advisory to mitigate a vulnerability

If a vulnerability is found, and CSIRT members of the SVG see the problem and wish to take operational mitigation, then they may. Also, the SVG and CSIRT may discuss taking operational action, particularly for e.g. 'High' risk issues which are difficult to fix and fairly straightforward to mitigate.

## 8.5 CSIRT Team will be informed when advisories are issued

SVG will inform CSIRT whenever it issues an advisory.

## 8.6 CSIRT Team will be informed of issues which cannot be fixed

This may be because there isn't a practical way of fixing it, or the problem is part of the design of the system. In this case CSIRT will be informed, with recommendation of any mitigating action that should be taken or problems they should be alert to.

## 8.7 CSIRT Team may consult the SVG RAT

CSIRT Team may see the RAT as a resource and consult the RAT where appropriate. This may include if the IRTF is investigating an incident and they wish the RAT to investigate some of the software. This may also include a request for an opinion on a vulnerability which is not part of the EGI UMD middleware. In general, the CSIRT Team and the SVG will work together to ensure that any possible problem is investigated, and the deployed infrastructure is sufficiently secure.

## 8.8 CSIRT Team members may join the CSIRT sub-group

Any CSIRT team member may join the CSIRT sub-group, which handles non-UMD software, if they wish. It was found that adding the whole team meant that a large number of people who are not interested in handling these were added to the issues. This CSIRT sub-group can be considered to be a RAT extension for issues that are not confined to UMD middleware.

# 9 NGI/SITES VIEW AND RESPONSIBILITIES

## 9.1 NGIs and Sites will receive advisories

Advisories are sent to NGI security contacts and Site security contacts. These lists are populated from the EGI GOCDB, using details entered for all certified sites.

Advisories are issued publicly on the EGI Wiki at

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

## 9.2 Sites should install up to date software

Sites should ensure that software is up to date, including installing up to date versions of the middleware distributed by the EGI/UMD and take note of appropriate advisories. Sites should be aware that CSIRT has the power to suspend sites from the infrastructure if they fail to apply critical security updates, however they should be given due warning and instruction on appropriate action to take to avoid suspension.

## 9.3 Sites should report any vulnerabilities they find

If a site finds a vulnerability, it should be reported as described in section 4.

## 9.4 NGIs and sites are invited to join the SVG

The RAT is drawn from both Sites and NGIs, and software providers. NGIs and sites are invited to provide a RAT member, the workload induced on a RAT member should only take a small portion of that persons time. An incentive to provide membership is the opportunity to influence the process as well as helping to provide a secure infrastructure.

# 10 NOTES AND EXCEPTIONS

Some cases occur which don't neatly fit into this process, and exceptions occur. Generally the rule is to use some common sense, and make sure that information is not exposed that is useful to an attacker, yet sites are alerted to problems and can take necessary action. Here are some examples of situations and suitable ways of handling them.

## *10.1 Collaborating Projects still vulnerable*

If software which fixes a vulnerability is available for installation on the EGI, but not for a collaborating project then the Advisory is not placed on the wiki, but just sent to the e-mail contacts with distribution set as "Community Wide Distribution Allowed" rather than the usual "unlimited distribution allowed"

## *10.2 Multiple problems in 1 piece of software*

If several vulnerabilities are found in 1 piece of software, e.g. from a vulnerability assessment, then 1 entry in the tracker, (and when they are fixed 1 advisory) is adequate. It is thought that 1 advisory concerning multiple (often 'Low' risk problems) with 1 piece of software is better than multiple advisories concerning 1 piece of software.

## *10.3 A problem that affects software from more than 1 source*

If a problem is found, which requires a new version of software from more than 1 distinct provider then each should be a separate ticket, and a separate advisory should be issued. This is not mandatory, but preferred.

## *10.4 Operational action is taken to mitigate the risk*

If CSIRT issues an advisory, for example to mitigate a 'High' risk vulnerability, this is usually sent to members of the project and not made public. If this substantially reduces the Risk the TD may be set as that of a lower risk category, if CSIRT considers this acceptable.

# 11 REFERENCES


| | |
|---|---|
| **R 1** | The (EGEE) Grid Security Vulnerability Group – Process and Risk Assessments for Specific Issues<br>https://edms.cern.ch/document/977396 |
| **R 2** | The EGI InSPIRE Incident Response Procedure<br>https://documents.egi.eu/secure/ShowDocument?docid=710 |
| **R 3** | The EGI InSPIRE Software Vulnerability Issue handling Procedure MS405<br>https://documents.egi.eu/secure/ShowDocument?docid=47 |
| **R 4** | The EGI CSIRT Critical Vulnerability Operational Procedure<br>https://documents.egi.eu/public/ShowDocument?docid=283 |