World of Code Storage Architecture Upgrade Investigation

VICTOR HAZLEWOOD, University of Tennessee, Knoxville, USA
UTKARSH PRATIUSH, University of Tennessee, Knoxville, USA
JON CALVIN WETZEL, University of Tennessee, Knoxville, USA

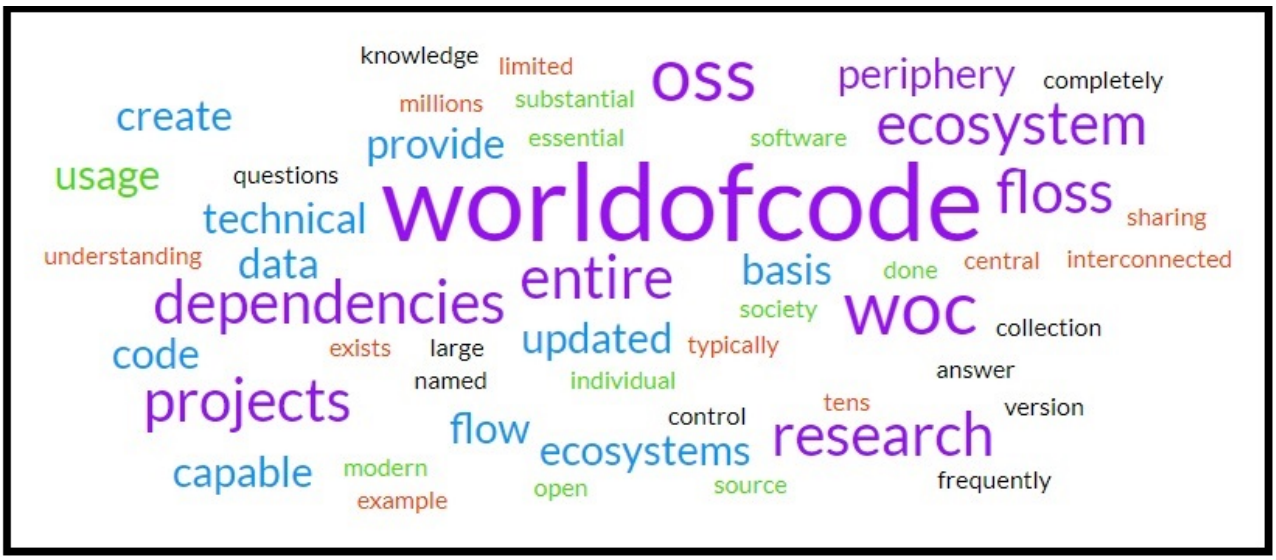

Fig. 1. World of Code Abstract Word Cloud

This study investigates enhancements to the storage architecture of the World of Code (WoC) system, aimed at efficiently supporting expansive datasets from the open-source software community. Specifically, the paper evaluates the feasibility and security implications of mounting the \lustre\isaac and \lustre\utstorr file systems via NFS on the WoC infrastructure. It conducts a thorough security assessment based on UTK baseline security controls, which resulted in approximately fifty questions and culminated in eight actionable recommendations. The analysis concludes that the risk associated with these mounts is low, primarily when achieved via bindfs on the da5 server. The successful implementation showcases the potential for other projects to partially satisfy their storage needs through the University of Tennessee's High Performance & Scientific Computing (HPSC) facilities. This approach not only extends storage capabilities but also enhances security, efficiency, and cost-effectiveness.

CCS Concepts: • **Open Source Software**; • **Software Engineering**;

Additional Key Words and Phrases: Open Source Software, Software Engineering, Data Intensive

ACM Reference Format:
Victor Hazlewood, Utkarsh Pratiush, and Jon Calvin Wetzel. 2024. World of Code Storage Architecture Upgrade Investigation. In . ACM, New York, NY, USA, 13 pages. https://doi.org/XXXXXXX.XXXXXXX

1 INTRODUCTION

The NSF sponsored World of Code (WoC) [1] [2] [3] has created a research infrastructure that curates a collection of source-code and version control history data from an extremely large portion of the free/libre and open-source software (FLOSS) ecosystem to enable a capability for the research community to study this ecosystem. WoC provides this curated data to enable the research community "to study FLOSS at large and answer both theoretical and practical questions that rely on the availability of the entirety of FLOSS data" [4]. Much of this source-code and version control history information is downloaded, stored, transformed, reorganized, and results stored using the resources of UTK's ISAAC Next Generation (ISAAC NG) cluster and a collection of nine WoC servers that we will call the WoC system.

CS540 Advanced SoftWare Engineering, April 2024, Knoxville, TN

ACM ISBN 978-x-xxxx-xxxx-x/YY/MM…$15.00
https://doi.org/XXXXXXX.XXXXXXX

Due to the vast size of FLOSS and the intention of WoC to download, store, and process monthly snapshots of the millions of FLOSS projects, the amount of storage resources that can be consumed to transform and provide the version control history information results is at a petabyte scale which would classify WoC as a Big Data application [5]. With this knowledge of WoC being in the Big Data class, it is worth investigating the data flow, use of storage, and possibilities to augment the storage architecture of WoC for storage efficiencies ultimately leading to improved data management and reduced cost. The storage available for WoC includes a 300 terabyte quota on the 3.6 petabyte Lustre high performance parallel file system, named \lustre\isaac, from the ISAAC NG cluster, a 200 terabyte quota on the 2.7 petabyte Lustre file system, named \lustre\haven, from the ISAAC Legacy cluster, and the approximately 1.2 petabytes of storage available and spread across the nine servers that make up the WoC system. Two new capabilities that might be of use to WoC is to mount \lustre\isaac directly on one or more of the WoC system servers and also to mount the new University of Tennessee Storage resource for Research (UT-StorR) long term archival storage system which became operational in April 2024. The University of Tennessee, Knoxville (UTK) Office of Innovative Technology (OIT) High Performance & Scientific Computing (HPSC) group manages the ISAAC NG cluster for use by the University research community and provisioned and provides UT-StorR for use in the long term storage of research data.

This work proposes the following questions:

RQ: Is it possible to mount \lustre\isaac via NFS on one or more of the WoC servers?

RQ: Is it possible to mount \lustre\utstorr via NFS on one or more of the WoC servers?

RQ: What are the security implications to mount \lustre\isaac and \lustre\utstorr on one or more of the WoC servers?

Our work attempts to answer the stated research questions which we believe may ultimately lead to efficiencies in data management of WoC and also longer term cost reduction for storage costs. The following sections describes the project methodology, the security assessment including a brief summary of the assessment report, a description of the file system mounting, conclusions, and future work. This work also includes a copy of the full assessment report in the Appendix that describes the security assessment methodology used, controls assessed, findings, and recommendations to WoC system administrators to meet a minimum standard for security for mounting the Lustre file systems from OIT HPSC resources onto WoC servers along with some other security related recommendations. Due to the sensitive nature of the information contained in the full assessment report, it is recommended that the security assessment report should not be published.

2 METHODOLOGY

The WoC storage architecture upgrade investigation methodology involves four steps itemized below:

- WoC Investigation: Read WoC paper and review WoC storage to investigate and document data flow
- WoC Storage Resources: Review WoC storage capabilities and usage on WoC servers and ISAAC clusters
- WoC System Security Assessment: ISAAC sysadmins requested a security assessment prior to any ISAAC file system NFS mounting. An assessment was performed
- File System Mounting: Work with ISAAC sysadmins to review assessment, address any concerns, and mount as appropriate

The investigation step involved getting familiar with WoC including reading the 2021 paper and an inspection of the storage in use on the ISAAC clusters and WoC systems in order to determine data flow for WoC. The data flow is described in Figure 2 which came from the WoC paper [1]. The second step involved going into the details of the WoC systems equipment and storage capabilities and current usage to map the data flow to the resources as shown in Figure 3. Figure 4 shows the equipment involved in the compute and storage resources of WoC. At the request of the ISAAC system administrators a security assessment which was step three and a security assessment was performed to determine the security posture of the WoC systems and the level of risk if mounting of \lustre\isaac and \lustre\utstorr are to be performed. The final step, the mounting, would be performed if the assessment determined the risk to be low from the ISAAC campus clusters perspective.

3 WOC DATA FLOW

The WoC paper [1] provides a high level overview of the WoC data flow shown in **Figure 2**. **Figure 3** shows our understanding of the mapping of that data flow to the systems and storage resources available to WoC. The initial data from the source code-forge (github, Bitbucket, etc.) is stored on the Lustre file systems on the ISAAC clusters and processed to generate the Commit, Tree, Blob, Author, and other information and then that data is reorganized and stored in various locations on the WoC systems. The servers and associated file systems for WoC data storage are listed in **Figure 4** which is a table of the servers, some system information, the storage file system names, and the amount of storage available and used as of April 10, 2024.

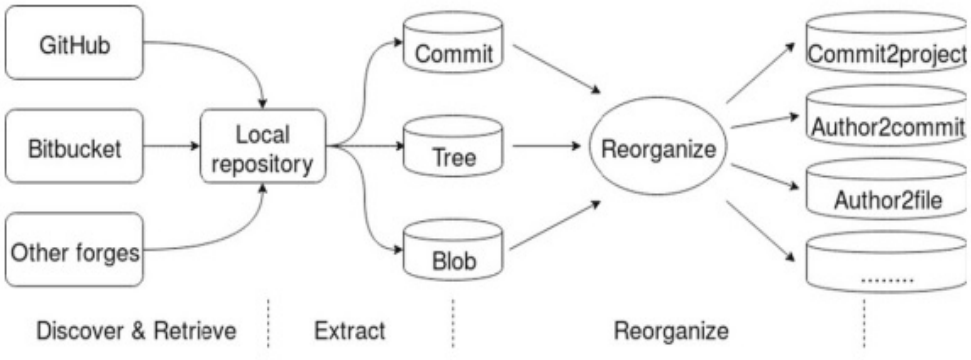

Fig. 2. World of Code data flow

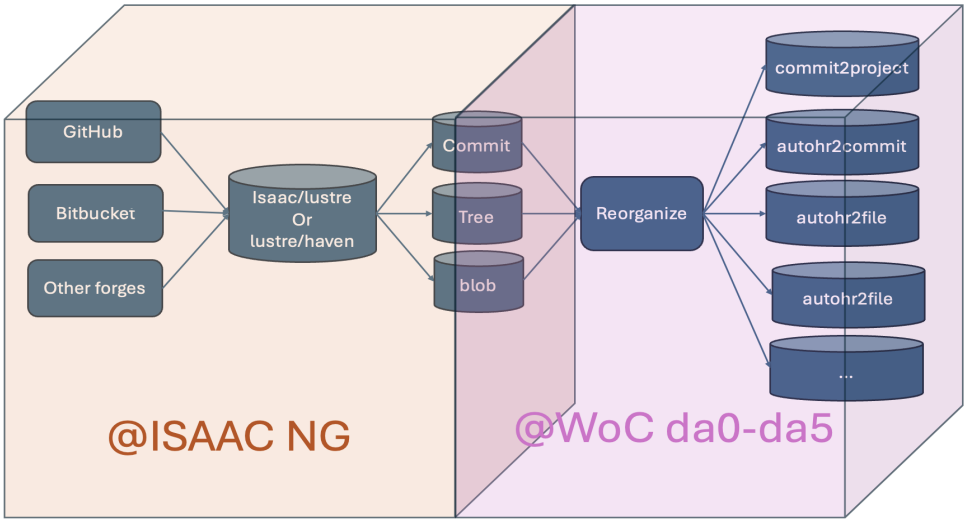

Fig. 3. Proposed World of Code data flow

Server	OS	Memory GB	Storage TB	Storage Used TB	File System	Server	Cores
DA0	RHEL 7.9	384	35	31	/export/data	R720XD	12 Sandy Bridge
DA1	RHEL 7.9	384	35	33	/export/data	R720XD	12 Sandy Bridge
DA2	RHEL 7.9	384	35	35	/export/data	R720XD	12 Sandy Bridge
DA3	RHEL 7.9	384	70	61	/export/data	R730XD	8 Haswell
DA4	RHEL 7.9	768	90	77	/data	R730XD	8 Haswell
DA5	RHEL 7.9	1280	123	115	/export/data	R740AD	40 Skylake
DA6	RHEL 7.9	256	90	77	/data	C4140	40 Skylake
DA7	Ubuntu 20.04.6	256	1700	464	/mnt/corrino/... (11)	AOM-S3616	16 Cascade Lake
DA8	Ubuntu 22.04.3	384	1700	306	/mnt/ordos/... (6)	AOM-S3616	16 Cascade Lake
WoC Systems				**1199**			
ISAAC Legacy	CentOS 7.6.1810	-	200 (quota)	192	/lustre/haven/...	-	-
ISAAC Next Gen	RHEL 8.7	-	350 (quota)	293	/lustre/isaac/...	-	-
UT-StorR	-	-	800/6300	0	/lustre/ut-storr/...	-	-
ISAAC				**485**	/lustre/ut-storr/...	-	-

Fig. 4. World of Code data storage servers

4 SECURITY ASSESSMENT

The ISAAC campus clusters are available for use by the campus community to support the research and academic mission of the University of Tennessee. The high-performance parallel Lustre file systems available on the ISAAC Legacy and ISAAC NG clusters represent some of the largest, if not the largest, on premises storage subsystems and file systems available to the campus research community. In efforts to provide more services and capabilities for the research community and to lower the overall cost of storage to the university, the Lustre file systems of the ISAAC clusters are being used in new and novel ways to support the research community. One example is the \lustre\isaac file

system is now served as a Windows file share to the controller of the Illumina NovaSeq 6000 and MiSeq DNA sequencing instruments of the UTK Genomics Center which is a UTK research Core Facility [6]. This eliminates the need for that core facility to have to maintain their own storage for the operation of these two instruments for the Center. Providing access to the Lustre file systems to researcher computer systems is a capability that OIT HPSC wants to investigate and to determine how to safely and securely provide this type of access to these Lustre file systems (the landing pad for the UT-StorR long term archival storage is also a Lustre file system). The use of the Lustre file systems directly by WoC systems represents a significant step in this direction. In this pursuit the OIT HPSC system administrators required a security assessment to assess the security of the WoC systems and the risk (from the perspective of the ISAAC clusters and the Lustre file systems) of mounting any of the Lustre file systems on the WoC systems. The following subsections describe the assessment methodology and a summary of the results of the assessment with the full security assessment in the Appendix.

4.1 Security Assessment Methodology

The methodology used for the WoC system security assessment was to assess relevant controls from the University of Tennessee, Knoxville fifty-five baseline security controls [7] on the WoC systems following the NIST 800-53A publication [8] on assessing security and privacy controls, to provide WoC server system administrators a questionnaire with any outstanding security questions from the assessment, evaluate the findings from the assessment and from the responses to the questionnaire, and provide any recommendations for security posture improvement prior to mounting the ISAAC NG \lustre\isaac and \lustre\utstorr Lustre file systems. The fifty-five baseline controls were narrowed down to fourty-three controls that were identified that should be assessed. See the Appendix section A.3 for the list of the controls assessed. See the Appendix section A.4 for the information related to the findings that were observed or obtained from the assessment questionnaire.

4.2 Assessment Report Summary

The results of the security assessment described in detail in the Appendix determined the risk to be low for the mounting of the \lustre\isaac scratch directory of user audris as readonly and for the mounting of \lustre\utstorr for user audris as read/write. Additionally, Dr. Mockus requested only da5 to perform the client-side mount of \lustre\isaac. The only critical issue from the assessment was related to the uid and gid information. ISAAC uses HPSC ldap and WoC systems uses EECS ldap. Each ldap has user “audris” but with different uid and gid values. The uid and gid information from the HPSC ldap is audris: uid 8320 and primary group 3319. The uid and gid information from the EECS ldap is audris: uid 22923 primary group 2343. This was addressed by using bindfs to remount \lustre\isaac\audris as \mnt\audris_map with mapping the uid and gid and exporting only to the WoC server da5. The full assessment report is in the Appendix and provides the Methodology, Controls Assessed, Findings, and Recommendations. Other security recommendations besides those necessary to mount the Lustre file systems are described in the report and provided to the WoC administrators for review and consideration

5 FILE SYSTEM MOUNTING

Per the assessment, in order to use network file system (NFS) to mount \lustre\isaac\audris on da5 uid mapping for one uid and gid mapping for one gid needed to be performed. This was done by using the Linux bindfs which is a FUSE file system capability for mounting a directory to another location on a system and capabilities inside the mount point can be altered, such as, uids, gids, permissions, and other items. The host used on the ISAAC cluster to server as the NFS export server was lnet2.cn.isaac.utk.edu. Dr. Mockus did request group 2354 instead of his default group 2343 which was provided. The bindfs configuration used was:

```
bindfs --map=8380/22923:@3319/@2354 /lustre/isaac/scratch/audris /mnt/audris_map
```

This is equivalent to the following line in /etc/fstab on lnet2:

```
/lustre/isaac/scratch/audris /mnt/audris_map fuse.bindfs map=8380/22923:@3319/@2354 0 0
```

In addition, the following was added to /etc/exports on lnet2:

```
/mnt/audris_map 10.22.16.183/32(sync,ro,no_wdelay,fsid=123)
```

The NFS client configuration on da5 had this added to the \etc\fstab file with the mount point on da5 as \da9_data:

```
lnet2.cn.isaac.utk.edu:/mnt/audris_map \da9_data nfs4 ro,async,hard,nosuid 0 0
```

With all the configuration listed above being executed by the respective servers, then the client mount of \lustre\isaac\audris was provided to da5 from \mount\audris from the lnet2 server. The access to the mount from da5 is shown in Figure 5.

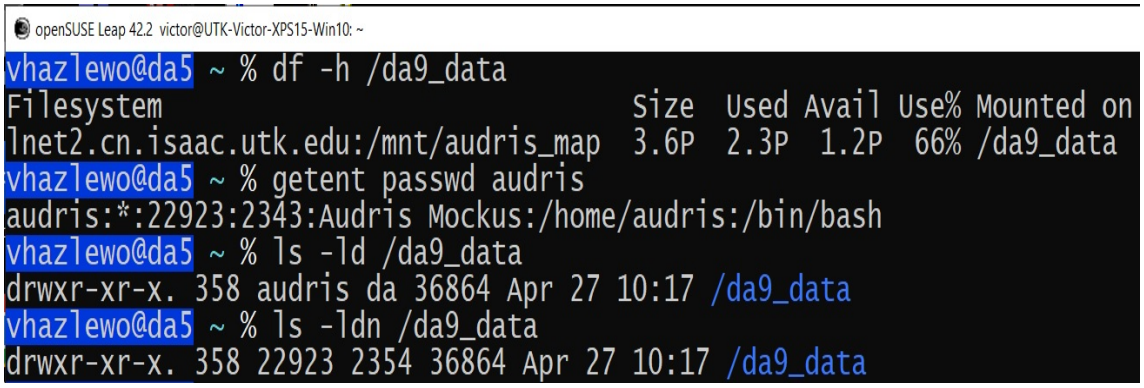

Fig. 5. Lustre mount on da5

6 CONCLUSIONS

It was determined that the risk to mount \lustre\isaac\audris read-only was low so mounting of that file system on da5 was configured on the ISAAC side on a system called LNET2 by the ISAAC system administrator and configured on da5 by Dr. Mockus with the client side fstab configuration sample provided by the ISAAC system administrator. Therefore, it was possible to mount an ISAAC Lustre scratch directory readonly with low risk and provide direct access to the WoC scratch directory files for use via NFS on WoC systems. **This simple case demonstrated Lustre mounting via NFS and possibilities for other projects!**

The team ran out of time for completing the mount for access to \lustre\utstorr as there is mandatory training required by OIT HPSC for access to UT-StorR. This was moved to future work for the OIT HPSC team.

7 FUTURE WORK

For NFS export of Lustre file systems from either of the ISAAC clusters and UT-StorR only the simpliest case was investigated in this work and the result was an exporting of a single directory as readonly that only had files owned by a single user and group ownership by a single group. Of course, many much more complicated scenarios exist and should be investigated in how uid and gid mapping can be done efficiently, effectively, and be performant when a directory would be NFS exported with a research group of two or more people. This would include determining the minimum security requirements to NFS mount ISAAC Lustre scratch and/or project directories read/write. How to efficiently map multiple uids and gids would also need to be determined. Due to time constraints the \lustre\utstorr file system was not mounted and the OIT HPSC group should continue that work to provide Dr. Mockus the required training and subsequently mounting of \lustre\utstorr on a WoC server(s) of Dr. Mockus' choice. The network bandwidth between the ISAAC NG cluster and UT-StorR system at the Kingston Pike Building data center and the data center in Min Kao that houses the WoC servers is unknown. That should be investigated and then subsequent benchmarking for NFS performance should be performed to make sure sufficient bandwidth is available, to determine if the NFS I/O would be expected to perform well, and what the possibly limitations may be. Lastly, it should be evaluated if read/write could be allowed securely to other researcher systems for more complex scenarios and demonstrate that WoC and other Big Data research projects could obtain storage from ISAAC, thereby, reducing the overall costs of storage since OIT HPSC buys storage in bulk for the ISAAC clusters.

ACKNOWLEDGMENTS

We wish to acknowlege and thank Dr. Audris Mockus and his graduate student Luis Gonzales Villalobos for help with this work and also the system administrator staff from the UTK OIT HPSC, George Butler and Matthew Bachstein.

A APPENDIX

A.1 World of Code Lustre Mount Security Assessment Report

Our team proposed for the CS540 Advanced Software Engineering project to analyze the data workflow and storage architecture of the World of Code (WoC) project and investigate potential improvements and/or efficiencies that may be obtained by mounting the ISAAC Next Generation (NG) cluster Lustre file system (\lustre\isaac) and the UT-StorR archival storage Lustre file system (\lustre\utstorr) directly onto World of Code servers. The Office of Information Technology (OIT), High Performance & Scientific Computing (HPSC) group required a security assessment of the WoC systems to assess the system for risk, security vulnerabilities, identity and access management (IAM), and other security related items prior to mounting the WoC project directory from the ISAAC NG and UT-StorR onto WoC system(s). This assessment report describes the methodology used, controls assessed, findings, and recommendations to WoC system administrators to meet a minimum standard for security for mounting the Lustre file systems from OIT HPSC resources onto WoC servers.

A.2 Methodology

The methodology employed for the WoC system security assessment was to assess relevant controls from the University of Tennessee, Knoxville fifty-five baseline security controls [7] on the WoC systems following the NIST 800-53A publication [8] on assessing security and privacy controls, to provide WoC server system administrators a questionnaire with any outstanding security questions from the assessment, evaluate the findings from the assessment and from the responses to the questionnaire, and provide any recommendations for security posture improvement prior to mounting the ISAAC NG \lustre\isaac and \lustre\utstorr Lustre file systems.

A.3 Controls Assessed

The fifty-five controls in the UTK baseline security controls were evaluated by the project team and some were determined to be irrelevant for the purposes of this assessment. The description of the security control and any supplemental information is listed in NIST 800-53 Appendix F Security Control Catalog [9] and the assessment criteria is listed in the NIST 800-53A publication [8]. The controls assessed were the following:

- AC-2 Account Management
- AC-3 Access Enforcement
- AC-5 Separation of Duties
- AC-7 Unsuccessful Login Attempts
- AC-17 Remote Access
- AC-18 Wireless Access
- AT-2 Security Awareness Training
- AT-3 Role-Based Security Training
- AU-6 Audit Review, Analysis, and Reporting
- CA-2 Security Assessments
- CA-3 System Interconnections
- CA-7 Continuous Monitoring
- CM-2 Baseline configuration
- CM-3 configuration change control
- CM-4 security impact analysis
- CM-6 configuration settings
- CM-7 least functionality
- CM-8 information system component inventory
- CM-9 configuration management
- CM-10 software usage restrictions
- CP-2 Contingency plan
- CP-9 Information System Backup
- IA-4 Identifier Management
- IA-5 Authenticator Management
- IR-4 Incident Handling
- PL-2 system security plan
- PL-4 rules of behavior
- PS-3 personnel screening
- PS-4 personnel termination
- PS-7 third party personnel security

- PS-8 personnel sanctions
- RA-5 vulnerability scanning
- SC-5 Denial of Service Protection
- SC-7 Boundary Protection
- SC-12 Cryptographic Key Establishment and Management
- SC-13 Cryptographic protection
- SC-20 Secure Name/Address Resolution Service (authoritative source)
- SC-21 Secure Name/Address Resolution Service
- SC-22 Architecture and provisioning for Name/Address Resoution Service
- SI-2 Flaw Remediation
- SI-3 Malicious Code Protection
- SI-4 Information System Monitoring
- SI-5 Security Alerts, Advisories, and Directives

The controls determined to be irrelevant and not related to an assessment for an external mount of a file system from the fifty-five security controls in the UTK baseline security controls [1] were the following:

- CA-5 Plan of Action and Milestones
- SI-8 SPAM protection
- IA-6 Authenticator Feedback
- MP-2 media access
- MP-3 media marking
- MP-4 media storage
- MP-5 media transport
- MP-6 media sanitization
- MP-7 media use
- SA-8 Security Engineering Principles
- SC-15 Collaborative Computing Devices
- SI-12 Information Handling and Retention

A.4 Findings

The WoC paper [1] provides a high level overview of the WoC data flow shown in **Figure 2**. **Figure 3** shows our understanding of the mapping of that data flow to the systems and storage resources available to WoC. The initial data from the source code-forge (github, Bitbucket, etc.) is stored on the Lustre file systems on the ISAAC clusters and processed to generate the Commit, Tree, Blob, Author, and other information and then that data is reorganized and stored in various locations on the WoC systems. The servers and associated file systems for WoC data storage are listed in **Figure 4** which is a table of the servers, some system information, the storage file system names, and the amount of storage available an used as of April 10, 2024.

A.4.1 Access Management and Identification and Authentication Families. The user identity information comes from the EECS department ldap server and local entries in the /etc/passwd and /etc/group files. The authentication can come from the University Linux ldap server to pick up NetID, password, and Duo authentication, however, SSH key authentication is enabled for a large number of users (if not all of them) for access to the WoC systems. Access is controlled by the system administrator enabling and/or disabling users having access to the system. Remote access is provided by the SSH service which encrypts each session. **da0, da2, da3, da5, da6, da7, da8 are accessible from the public internet**. da1 and da4 were not observed to be accessible via SSH on port 22 by direct attempt.

The relevant ldap information for Dr. Mockus in the HPSC ldap and in the EECS ldap is shown below. The usernames are the same and the uidNumbers are different. Group names and gid numbers do not match in the two different ldap domains.

EECS ldap:
dn = uid=audris,ou=People,dc=cs,dc=utk,dc=edu
objectClass = ['utkcsAccount', 'account', 'posixAccount', 'inetLocalMailRecipient', 'top']
uid = audris
seeAlso = uid=audris,ou=People,ou=Knoxville,dc=tennessee,dc=edu
Primary Group = 2343 (audris)
gecos = Audris Mockus
homeDirectory = /home/audris

[1]These were determined by Victor Hazlewood who is a Certified Information System Security Professional since August 2002.

loginShell = /bin/bash
userPassword = 'SASLaudris'
uidNumber = 22923
cn = Audris Mockus
HPSC ldap:
dn: uid=audris,ou=People,dc=hpsc,dc=tennessee,dc=edu
objectClass: person
objectClass: organizationalPerson
objectClass: inetOrgPerson
objectClass: posixAccount
objectClass: top
objectClass: shadowAccount
objectClass: hostObject
uid: audris
gecos: Audris Mockus
uidNumber: 8380
homeDirectory: /nfs/home/audris
loginShell: /bin/bash
host: isaac
gidNumber: 3319

User accounts are reviewed periodically and accounts are disabled that have not been used in 360 days. A cron job emails administrators the list of users who have access and not used the system in 360 days and those accounts need to be addressed manually by the system administrators. There are no shared accounts but there is a group called woc that users are added to from classes or research purpose. There are only user and administrator roles. Only Luis Villalobos and Dr. Mockus have administrator role and privileges.

Only SSH protocol is used for remote access which allows SCP and SFTP. da0 was running OpenSSH_7.2p2, OpenSSL 1.0.2j-fips from 26 Sep 2016. SSH key-based authentication is enforced, making WoC not susceptible to brute-force password guessing attacks. Default Rate-Limiting mechanisms at the firewall level (standard for RHEL) limit the number of connection attempts from a single IP address within a specific time frame.

A.4.2 Awareness and Training Family. The system administrators do perform their annual UTK security awareness training. There is no role based security training specifically for WoC sytem administrators.

A.4.3 Audit and Accountability Family. There was no indication from the answers to the questionnaire that log information of the WoC systems are reviewed or analyzed periodically for indications of inappropriate or unusual activity. Only the standard audit was mentioned as available to allow monitoring and logging of security-related events. It was indicated that SELinux was enabled but the /etc/sysconfig/selinux file was empty on da0, da7 (ishia), da8 and the other systems had different configurations for selinux (shown below). That file would have a section that shows you whether SELinux is in permissive mode, enforcing mode, or disabled, and which policy is supposed to be loaded.

From review of each of the WoC systems /etc/rsyslog.conf file, each of the WoC systems log locally just to themselves and there is no centralized log server, nor are the log files sent to a University or EECS security operations center or intrusion detection service.

The following is the contents of the /etc/sysconfig/selinux file on da2, da5

```
# This file controls the state of SELinux on the system.
# SELINUX= can take one of these three values:
#     enforcing - SELinux security policy is enforced.
#     permissive - SELinux prints warnings instead of enforcing.
#     disabled - No SELinux policy is loaded.
SELINUX=enforcing
# SELINUXTYPE= can take one of three two values:
#     targeted - Targeted processes are protected,
#     minimum - Modification of targeted policy. Only selected processes are protected.
#     mls - Multi Level Security protection.
SELINUXTYPE=targeted
```

The following is the contents of the /etc/sysconfig/selinux file on da3

```
# This file controls the state of SELinux on the system.
# SELINUX= can take one of these three values:
```

```
#     enforcing - SELinux security policy is enforced.
#     permissive - SELinux prints warnings instead of enforcing.
#     disabled - No SELinux policy is loaded.
#SELINUX=enforcing
SELINUX=permissive
# SELINUXTYPE= can take one of three two values:
#     targeted - Targeted processes are protected,
#     minimum - Modification of targeted policy. Only selected processes are protected.
#     mls - Multi Level Security protection.
SELINUXTYPE=targeted
```

The following is the contents of the /etc/sysconfig/selinux file on da6

```
# This file controls the state of SELinux on the system.
# SELINUX= can take one of these three values:
#     enforcing - SELinux security policy is enforced.
#     permissive - SELinux prints warnings instead of enforcing.
#     disabled - No SELinux policy is loaded.
SELINUX=enforce
# SELINUXTYPE= can take one of three values:
#     targeted - Targeted processes are protected,
#     minimum - Modification of targeted policy. Only selected processes are protected.
#     mls - Multi Level Security protection.
SELINUXTYPE=targeted
```

Note "SELINUX=enforce" is not a valid value for SELINUX.

A.4.4 Security Assessment and Authorization Family. Security assessments are different from vulnerability scanning. Responses to the security assessment questions provided answers more suitable for vulnerability assessments. The answer to the continuous monitoring question was as follows:

Our continuous monitoring strategy is based on the following points:

(1) Establish Monitoring Goals and Objectives: Maintain the availability and performance of the WoC infrastructure. Identify security incidents, identify vulnerabilities, and ensure compliance with security policies and regulations per the EECS department.
(2) Identify Critical Assets and Data Sources: This includes servers (hardware), network devices, databases, applications, and logs.
(3) Select Monitoring Tools and Technologies: This is limited to vulnerability scanners (Security Center/Nessus) and future intrusion detection systems (IDS) implementation. We also utilize Grafana as an all-in-one performance and status monitoring service.
(4) Define Monitoring Metrics and Indicators: We defined key performance indicators (KPIs) for performance, incident detection, and response times.
(5) Implement Automated Monitoring and Alerting: We have configured monitoring tools to automatically collect and analyze performance data in real-time or near-real-time and set up alerting mechanisms to notify security personnel of events that require attention or investigation. We plan to incorporate security monitoring soon.
(6) Perform Regular Vulnerability Scanning: We plan to conduct regular vulnerability scans of the DA systems and applications to identify known vulnerabilities, misconfigurations, and weaknesses that attackers could exploit. We will integrate vulnerability scanning results into the continuous monitoring to promptly prioritize and remediate security issues.
(7) Analyze Security Logs and Events: We plan to analyze logs and security events generated by DA systems, applications, and network devices to detect signs of unauthorized access, malicious activities, and security breaches.
(8) Monitor User and Entity Behavior: We actively monitor the servers' performance. This monitoring leads to investigating user activities, behavior, and access patterns to detect insider threats, account compromise, and unauthorized activities that hinder the DA cluster's performance.
(9) Review and Respond to Security Alerts: With the future implementation of automated monitoring, we plan to regularly review security alerts generated and respond to security incidents (not only for performance) according to predefined procedures and escalation paths.
(10) Continuous Improvement and Evaluation: We continuously evaluate (at the end of each quarter) and refine the monitoring strategy based on lessons learned, feedback, and changes in the threat landscape.

No external system interconnections were observed besides the cross-mounting of file systems on the WoC systems and there were none described in the answers.

A.4.5 Configuration Management Family. Dell service tag information was provided by the EECS IT staff and provided information for the CM-8 information system component inventory information. From answers from the questionnaire, configuration is based on the template set up by EECS administrators. An answer to a configuration management question was stated that "all systems have precisely the same configuration." However, the selinux configuration on each system shows at least one important example that the configuration on each of the WoC systems are not the same. Four of the other configuration management questions were answered with Not Applicable for items like if there is a process for change management and if a configuration management tool is employed on the da systems. A tool and process for configuration management and change management was not observed to be in place nor answers provided from the questionnaire to indicate these important items were in place for WoC systems. It was stated that only essential services are enabled and unnecessary ports, protocols and services are disabled. OS software is only installed by the system administrators. There are no backups of the OS or configuration of the WoC systems.

A.4.6 Contingency Planning Family. The WoC servers and administrators will rely on the EECS IT staff and the University's OIT staff to help with any incident response and contingency. It was stated that WoC is a best-effort service and no contingency plan was provided.

A.4.7 Identification and Authentication Family. User identifiers (usernames) come from the EECS ldap. Authenticators (passwords) generally are ssh keys set up from class rosters or participants in WoC hackathons. No 2FA capability is generally used though it is available. Authenticators (passwords and ssh-key passwords) are obscured when used (via SSH).

A.4.8 Incident Response Family. The WoC servers and administrators will rely on the EECS IT staff and the University's OIT staff to help with any incident response and contingency. WoC administrators do not have automated detection at the OS level, but for the WoC website, they stated that they do. The incident response plan entails an automated email that gets generated to notify the administrators. The administrators will then perform investigation and resolution. WoC administrators maintain bi-weekly meetings where they discuss how to improve and better prepare for any future problems from lessons learned.

A.4.9 Planning Family. The information on WoC systems is all open research information. There is no written security plan. University policies apply for rules of behavior.

A.4.10 Personnel Security Family. WoC systems are available to students, the administrators and Graduate and post-doctoral researchers who are attempting to leverage WoC for research and publications. Access can be provided to people for research purposes by filling out a google form on the worldofcode.org website. When system administrators graduate, terminate employment, or terminate school work sudo privileges are removed. Accounts are disabled after 360 days.

A.4.11 Risk Assessment Family. In the answer to CA-2 from the questionnaire it was stated that vulnerability scanning was done quarterly but in the answer to CA-7 it was said that "... we plan to conduct regular vulnerability scans ..." and that "...we plan to analyze logs and security events generated by the da systems." There is inconsistency in responses to whether vulnerability scanning is performed or not. No vulnerability assessment report was requested, mentioned, or provided.

If regular vulnerability assessment are conducted then known vulnerabilities related to critical CVEs would likely be identified and could be mitigated. For example, a quick review of level 10 critical vulnerabilities for Red Hat Enterprise Linux indicated CVE-2018-14618 which states that curl before version 7.61.1 are vulnerable. The curl software on the WoC systems da0, da1, da2, da3, da4, and da6 are version 7.29.0 (a 2013 version) and the curl software on da5 is version 7.51.0 (a 2016 version) both have critical level 10 CVE vulnerabilities. The curl software on da7 is version 7.68.0 and the curl software on da8 is version 7.81.0 both of which are not vulnerable. This also refutes the statement from the answers to the questionnaire for WoC systems that "All systems have precisely the same configuration." It is likely that some software was needed to be installed on one or more of the WoC systems and along with those software were some dependencies on curl which caused them to be upgraded separately on da5, da7 and da8. Vulnerabilities of curl are all described at https://curl.se/docs/vuln-7.29.0.html. Vulnerability assessments and mitigation (called flaw remediation in the NIST 800-53 security controls) combined with configuration management as described in the NIST 800-53 document are not observed to be followed and the implementation of these controls provides a solid foundation for a good security posture.

A.4.12 System and Communication Protection Family. The WoC systems are protected by a firewall that restricts access to services with a deny all allow by exception capability. Crypto keys are generated locally. Researchers outside of UTK

are provided access to use WoC systems for research purposes. All WoC servers and the website hosted on them are accessible from the Internet. It was stated “If one of the WoC servers is compromised, it would be pretty easy to move to the others since we have listed instructions on how to do this on our tutorial website—although this requires key generation."

A.4.13 System and Information Integrity Family. There is no flaw remediation (vulnerability management) procedure in place on the WoC systems. The answer regarding Flaw Remediation was “we use standard RHEL process for updates, but that doesn’t address WHEN the updates are performed after a serious or critical flaw is found in RHEL software either by a vulnerability assessment report or by participating in a security alert or advisory capability. The answer regarding malicious code protection was “No, but SELinux is set to enforcing mode." however, enforcing mode is not set on all WoC systems. See section 4.3.

Regarding Information System Monitoring it was stated that “We monitor the servers’ health and performance through a Grafana dashboard. This dashboard allows the administrators to see performance issues with a particular DA system, which triggers a more in-depth investigation into what is hindering the server’s performance." The question was not about performance monitoring but more about unauthorized activity monitoring (related to SI-4 Information System Monitoring).

There is no indication on the systems or in the questionnaire responses that the WoC systems log information is monitored on a regular basis to detect potential attacks, successful or unsuccessful intrusions, or unusual activity warranting investigation.

A.5 Recommendations

This security assessment is intended to evaluate the WoC systems for suitability for mounting /lustre/isaac/scratch/audris from the ISAAC Next Generation cluster. Discussions with Dr. Mockus led to his specifically requesting the mount only on da5.eecs.utk.edu. The main issue discovered in the assessment was the IAM issues with user id and group id mapping between the two systems since they use different ldap IAM systems (see section 4.1). The ISAAC NG system administrator, George Butler, evaluated the situation and recommended use of bindfs to remount the file system and map the userid and gid on the server side to substitute to the required uid/gid. With the configuration capability with bindfs to map the uid/gid information and serve the directory via network file system as readonly, it is determined to be low risk to mount that directory readonly from ISAAC NG onto any WoC systems including the requested da5 system.

The assessment was performed for a specific purpose related to WoC and access to the ISAAC NG Lustre file system and as mentioned previously if mounting readonly and uid/gid mapping are put in place the risk will be low and as such the recommendations below that are critical will need to be put in place to ensure security for that purpose. Additional security findings are worth mentioning with corresponding recommendations. The recommendations will be listed as Critical for those necessary to precede the mounting of the requested file systems. The High, Moderate, and Low recommendations will be for Dr. Mockus to consider to improve the overall security posture of the WoC systems.

A.5.1 Critical Recommendations.

[**Critical**] To keep the risk low for mounting Dr. Mockus’ ISAAC NG lustre scratch directory to WoC systems the configuration by OIT HPSC staff of the NFS server-side service should export the NFS as readonly and must include userid mapping to map HPSC ldap uidNumber 8380 to EECS ldap uidNumber 22923 and map HPSC group gid 3319 to EECS primary group gid 2343. HPSC staff have determined that the use of the uid/gid mapping feature in bindfs will provide this capability.

A.5.2 High, Moderate, and Low Recommendations.

[Medium] The quarterly vulnerability scan should be performed with the Tenable vulnerability scanning capability provided by OIT instead of Nessus (the free vulnerability assessment tool). OIT has licensed the Tenable vulnerability management system for campus use. When Vulnerability scanning is performed all critical and high vulnerabilities should be addressed without delay which would likely include upgrading curl which is vulnerable as described in the Findings

[High] System Administrators should exclusively use authentication with a two factor authentication (2FA) mechanism, such as, the University’s Duo 2FA for all privileged escalation (root access via sudo) and this authentication should be separate and distinct from the system administrators ssh-key authentication to their user accounts. This will separate user access from privileged access and prevent intrusions from escalating to root privileges just by compromise of the system administrator’s user authentication credentials.

[High] The WoC system’s should centralize their logs to a single WoC centralized log server and also report all WoC logs to the OIT central log service (LogRhythm today and the Dell Security Operations Center in the future).

Implementation of either of those would provide important system security monitoring capabilities.

[High] A process and procedure should be put in place to regularly disable users after classes are over and after hackathons are completed at a much shorter time duration than 360 days.

[High] It is recommended to use or set up a bastion host that would be the intermediary for people to access the WoC systems from the Internet. If this is not practical or interferes with usability, then one WoC system should be set up as the sole system to be accessible from the Internet to reduce the external attack surface and to be used to access the other WoC systems. Essentially, one of the WoC systems taking the role of a bastion host.

[Medium] It is recommended that a configuration management process and tool, such as, ansible be used to manage the configuration of the WoC systems beyond the initial base operating system configuration. This will provide a capability for a small number of system administrators to move away from manual configuration management and to manage the nine WoC systems in an automated, efficient and organized manner allowing for configuration rules to be set up and to be consistent across all the systems, across subsets of the WoC systems, or on individual WoC systems. Along with this capability it is much easier to restore a system after a malicious intrusion, hardware failure, or accidental mishap if ansible configuration management or similar tool is used.

[Low] System Administrators should have a role based training to cover the do's and don'ts for system administrator role for WoC systems to address AT-3 Role-Based Security Training and discuss University policies that cover the PL-4 Rules of Behavior security control.

REFERENCES

[1] Yuxing Ma, Tapajit Dey, Chris Bogart, Sadika Amreen, Marat Valiev, Adam Tutko, David Kennard, Russell Zaretzki, and Audris Mockus. World of Code: Enabling a Research Workflow for Mining and Analyzing the Universe of Open Source VCS Data. *Empirical Software Engineering*, 26:1–42, October 2021.

[2] Audris Mockus. Collaborative Research: CCRI: New: World of Code (WoC): The Development of Curated Code Resource to Support Research in Software Engineering. https://www.nsf.gov/awardsearch/showAward?AWD_ID=2120429&HistoricalAwards=false, September 2021. National Science Foundation Awards, Accessed April 28, 2024.

[3] James Herbsleb. Collaborative Research: CCRI: New: World of Code (WoC): The Development of Curated Code Resource to Support Research in Software Engineering. https://www.nsf.gov/awardsearch/showAward?AWD_ID=2120323&HistoricalAwards=false, September 2021. National Science Foundation Awards, Accessed April 28, 2024.

[4] Yuxing Ma, Tapajit Dey, Chris Bogart, Sadika Amreen, Marat Valiev, Adam Tutko, David Kennard, Russell Zaretzki, and Audris Mockus. World of Code: Enabling a Research Workflow for Mining and Analyzing the Universe of Open Source VCS Data. *Empirical Software Engineering*, 26:2, October 2021.

[5] Ibrar Yaqoob, Ibrahim Abaker Targio Hashem, Abdullah Gani, Salimah Mokhtar, Ejaz Ahmed, Nor Badrul Anuar, and Athanasios V Vasilakos. Big data: From beginning to future. *International Journal of Information Management*, 36(6):1231–1247, 2016.

[6] Innovation Office of Research and Economic Development. Research, Innovation, and Economic Development Core Facilities. https://research.utk.edu/oried/core-facilities/, December 2014. Accessed May 4, 2024.

[7] UTK-OIT. UTK Security Program Plan Baseline Security Controls. https://tiny.utk.edu/MinimumControls, April 2024. Accessed April 28, 2024.

[8] National Institute of Standards and Technology. Special Publication 800-53a, Revision 4, Assessing Security and Privacy Controls in Federal Information Systems and Organizations: Building Effective Assessment Plans. https://csrc.nist.gov/pubs/sp/800/53/a/r4/upd1/final, December 2014. Accessed April 28, 2024.

[9] National Institute of Standards and Technology. Special Publication 800-53, Revision 4, Security and Privacy Controls for Federal Information Systems and Organizations. https://csrc.nist.gov/pubs/sp/800/53/r4/upd3/final, January 2015. Accessed April 28, 2024.