DIMS System Requirements Documentation

Release 2.9.0

David Dittrich

September 20, 2016

Contents

CHAPTER 1

Scope

1.1 Identification

This System Requirements document (version 2.9.0) describes the high-level requirements for designing and implementing the Distributed Incident Management System (DIMS).

1.2 System overview

DIMS is funded by the Department of Homeland Security under contract HSHQDC- 13-C-B0013. For more information, see the DIMS Operational Concept Description v 2.9.0. and contract referenced in Section *Referenced Documents*.

The primary mission objectives for the DIMS system are operational in nature, focused on facilitating the exchange of operational intelligence and applying this intelligence to more efficiently respond and recover from cyber compromise. The secondary mission objectives are to create a framework in which tools to support the primary mission objectives can more quickly and easily be integrated and brought to bear against advancing techniques on the attacker side of the equation.

The DIMS project is intended to take this semi-automated sharing of structured threat information, building on the success of the Public Regional Information Security Event Monitoring (PRISEM) project and leveraging an existing community of operational security professionals known as Ops- Trust, and scale it to the next level. The intent of this software project is to allow for near real-time sharing of critical alerts and structured threat information that will allow each contributing party to receive information, alerts and data, analyze the data, and respond appropriately and in a timely manner through one user-friendly web application, DIMS.

Working with the use cases defined by MITRE and PRISEM users, building the features necessary to simplify structured information sharing, and operationalizing these within these existing communities, will allow DIMS to fill existing gaps in capabilities and support existing missions that are slowed down today by many complicated, manual processes.

The changes to existing systems consists of seamless integration of the three current systems into a single web application that enables each system to contribute to the data warehouse of information concerning threats, alerts, attacks and suspect or compromised user terminals within the infrastructure. Additionally, the integrated systems will be able to share and retrieve data, visually observe alerts through color coded visual indicators, while retaining the existing functionality of the current system.

1.3 Document overview

The structure of this document has been adapted principally from MIL-STD-498 (see Section *Referenced Documents*). Following this section are:

- Section *Referenced Documents* lists related documents.
- Section *Requirements* specifies the requirements for each major DIMS CSCI component. It includes descriptions of the high-level CSCI components (per contract) along with such specific requirements including (but not limited to): capabilities; interfaces; data inputs/outputs; adaptations; environment; resources; software quality factors; and design and implementation constraints.
- Section *Notes* provides an alphabetical listing of acronyms and abbreviations used in this document.
- Section *License* includes the copyright and software license under which DIMS is being released.

CHAPTER 2

Referenced Documents

The following documents describe the DIMS project and provide background material related to tasking.

1. DIMS Operational Concept Description v 2.9.0
2. DIMS Architecture Design v 2.9.0
3. DIMS Test Plan v 2.9.0
4. HSHQDC-13-C-B0013, "From Local to Gobal Awareness: A Distributed Incident Management System," Draft contract, Section C - Statement of Work (marked up version)
5. MIL-STD-498, Military Standard Software Development and Documentation, AMSC No. N7069, Dec. 1994.
6. Aldridge, J. Targeted Intrusion Remediation: Lessons from the Front Lines, August 2012. Black Hat USA 2012 Presentation. https://www.mandiant.com/blog/black-hat-usa-2012-presentation-targeted-intrusion-remediation-lessons-front-lines/
7. Beebe, N. and Clark, J. G. A hierarchical, objectives-based framework for the digital investigations process. Digital Investigation, 2(2):147–167, 2005. http://faculty.business.utsa.edu/nbeebe/pubs/DIP%20Framework%20Journal%20Submission%20v4%20-%20FINAL%20JDI%20author%20copy.pdf
8. Bluehat1. New MAPP Initiatives, July 2013. BlueHat Blog. http://blogs.technet.com/b/bluehat/archive/2013/07/29/new-mapp-initiatives.aspx
9. Boyd, J. R. (Col.). Boyd's OODA "Loop" From "The Essence of Winning and Losing", 2008. Available at http://www.d-n-i.net/fcs/ppt/boyds_ooda_loop.ppt
10. Ciardhuain, S.O. An Extended Model of Cybercrime Investigations. International Journal of Digital Evidence, 3(1), Summer 2004. http://www.utica.edu/academic/institutes/ecii/publications/articles/A0B70121-FD6C-3DBA-0EA5C3E93CC575FA.pdf
11. Dittrich, D. PRISEM Analyst's Handbook, December 2013.
12. Dittrich, D. PRISEM System Administration Handbook, December 2013.
13. Dittrich, D. Advanced Incident Response Capabilities Supporting Collaborative and Cooperative Responses. Unpublished manuscript, April 2007.
14. Dittrich, D. On Developing Tomorrow's "Cyber Warriors". In Proceedings of the 12th Colloquium for Information Systems Security Education, June 2008. http://staff.washington.edu/dittrich/misc/cisse2008-dittrich.pdf
15. Dittrich, D.. On the Development of Computer Network Attack Capabilities. Unpublished manuscript, February 2008. This work was performed for the National Research Council under agreement D-235- DEPS-2007-001.
16. Dittrich, D. The Conflicts Facing Those Responding to Cyberconflict. In USENIX ;login: vol. 34, no. 6, December 2009. http://www.usenix.org/publications/login/2009-12/openpdfs/dittrich.pdf

17. Executive Office of the President. Homeland Security Presidential Directive 7: Critical Infrastructure Identification, Prioritization, and Protection. http://www.dhs.gov/xabout/laws/gc_1214597989952.shtm, December 2003.

18. Federal Emergency Management Agency. National Response Framework, January 2008. http://www.fema.gov/pdf/emergency/nrf/nrf-core.pdf

19. Gragido, W. Understanding Indicators of Compromise (IOC) Part I, October 2012. http://blogs.rsa.com/will-gragido/understanding-indicators-of-compromise-ioc-part-i/

20. Hamilton, M. and Dittrich, D. An overview of the Public Regional Information Security Event Management Project, December 2013.

21. Hutchins, E., Cloppert, M. and Amin, R. Intelligence-Driven Computer Network Defense Informed by Analysis of Adversary Campaigns and Intrusion Kill Chains. In 6th Annual International Conference on Information Warfare and Security. Lockheed Martin Corporation, December 2011. http://www.lockheedmartin.com/content/dam/lockheed/data/corporate/documents/LM-White-Paper-Intel-Driven-Defense.pdf

22. Khurana, H., Basney, J., Bakht, M. Freemon, M., Welch, V., and Butler, R. Palantir: A Framework for Collaborative Incident Response and Investigation. In IDtrust '09: Proceedings of the 8th Symposium on Identity and Trust on the Internet, pages 38–51, New York, NY, USA, April 2009. ACM. http://middleware.internet2.edu/idtrust/2009/papers/05-khurana-palantir.pdf

23. Ieong, R. S. C. FORZA - Digital forensics investigation framework that incorporate legal issues. Digital Investigation, 3(Supplement-1):29–36, 2006. http://www.dfrws.org/2006/proceedings/4-Ieong.pdf

24. Mandiant. Using Indicators of Compromise to Find Evil and Fight Crime, August 2011. http://www.us-cert.gov/GFIRST/presentations/2011/Using_Indicators_of_Compromise.pdf

25. Mandiant. APT1: Exposing One of China's Cyber Espionage Units, February 2013. http://intelreport.mandiant.com/Mandiant_APT1_Report.pdf

26. Microsoft Developer Network. Chapter 3: Workflow and Process. http://msdn.microsoft.com/en-us/library/bb833024.aspx

27. The Mitre Corporation. Standarizing Cyber Threat Intelligence Information with the Structured Threat Information eXpression (STIX), 2012. http://makingsecuritymeasurable.mitre.org/docs/STIX-Whitepaper.pdf

28. Richards, C. Briefings - Colonel John R. Boyd, USAF, November 2009. http://www.ausairpower.net/APA-Boyd-Papers.html

Requirements

Attention: Throughout this (and other) DIMS documents, requirements are identified symbolically. For example, the first requirement below is identified by the symbol **modetoggles**. In this document, the definition of the symbol is shown in double-brackets in the subsection heading where it is defined to make the symbol explicitly known to the reader (e.g., **:ref:‘modetoggles‘**) In other documents that must trace back to these requirements, the use of *intersphinx* links are used, which render the subsection heading and thus show the same symbol in double-brackets to clearly show how requirements are linked.

3.1 Required states and modes

3.1.1 [[modeToggles]] Mode toggles

Modes described in this section can be be toggled **on** or **off** from the user dashboard interface. When toggeled, DIMS subsystems will either be made aware of the change (e.g., through the AMQP message bus, by monitoring a distributed file system path, or by checking for the presense of a flag indicating the state of the toggle on starteup). If notification of the change is not made (in a “push” fashion), DIMS subsystems will need to actively poll for the change.

Since DIMS is designed to accumulate event logs, this is just a variation of the existing event log collection mechanisms built into DIMS. Test and/or debug logs will be tagged so as to separate them from the security event logs. A mechanism to purge logs that are no longer necessary should be supported. (This may be an optional setting when toggles are turned **off**.)

3.1.2 [[testMode]] Test Mode

The DIMS system shall be designed to support a test mode that generates information useful for tests at the system level as described in the DIMS Test Plan, section Test levels.

3.1.3 [[debugMode]] Debug Mode

The DIMS system shall be designed to support a debugging mode that allows generation of increasingly verbose logs that will assist in debugging.

3.1.4 [[demoMode]] Demonstration Mode

In order to demonstrate a live instance of the DIMS system, without exposing any sensitive information it may contain, the DIMS system should support a demonstration mode that loads specially prepared **demonstration** data. This data may be fabricated, manually anonymized and/or collected from honeypot systems that are outside of any sensitive network blocks. This mode will also be useful for teaching students how to become analysts.

During normal use of the DIMS system with live data and anonymization turned on, the user may chose to save interim search results or other analysis products from the data processing stream to a separate storage location for use in demonstration mode. This should include the ability to export all of the data in a single archive file for simplicity in building a library of demonstration data. This data can then be made available along with the DIMS software and deployment utilities so someone can easily bring up a demonstration instance with little or no manual intervention.

Note: For an example of what this would look like, see how MozDef or GRR Rapid Response work by building and running their respective Docker images as described in their documentation. DIMS will model these projects in production of a simple demo-mode deployment.

When in demonstration mode, the system should take the set of search parameters that are given to the user interface and generate a hash to save with the results. When in debug mode later, the set of search hashes can be used to either pre-populate the user interface, or just be used to compare with later searches done in Demonstration mode. When it is recognized that a pre-recorded search is being initiated, rather than send the paramters off to search processors, the system can retrieve the saved search results related to the search hash and present them to the user. This allows the system to appear to function as normal, but without having to fill the databases with fake data. (This also allows a production system to be used in demonstration mode without polluting the production databases.)

When done using demonstration mode, an additional option to save or delete any saved files should be supported to simulate multi-session and multi-user use of DIMS. When selecting deletion, there are two sub-states:

1. Selecting **clean all** will clean out all of the demonstration data and intermediary results, allowing a production system to be completely cleaned of any demostration related data;
2. Selecting **clean temporary files** only deletes the intermediary saved results, not the original demonstration data. This resets the demonstration back to the start for predictable repetition.

If it is easier, both of these sub-modes can involve complete deletion of a single demonstration partitioned datastore (only the first mode would immediately re-load the demonstration datastore from the original demonstration dataset.

3.2 CSCI capability requirements

The DIMS system is divided into the following high-level Computer Software Configuration Item (CSCI) sets, per the acquisition contract referenced in Sections *System overview* and *Referenced Documents*.

CSCI	Label	Contract Item
Backend Data Stores	BDS	C.3.1.1
Dashboard Web Application	DWA	C.3.1.1
Data Integration and User Tools	DIUT	C.3.1.2
Vertical/Lateral Information Sharing	VLIS	C.3.1.3

This subsection is divided into subparagraphs to itemize the requirements associated with each capability of these CSCI sets. Each capability is labelled with its specific CSCI

3.2.1 Backend Data Stores (BDS) CSCI

The following sections describe the requirements for the Backend Data Stores (BDS) CSCI.

[[attributeStorage]] Attribute Storage

The DIMS system must have the ability to store additional attributes for each user (such as which CIDR blocks they are responsible for protecting, which top level Domain Name System domains, and/or which high-level activities (e.g., campaigns) they wish to monitor. This capability allows the system to notify the user when there are messages or email threads of interest, and to facilitate providing regular tailored reports or alerts about activity of interest to them. These attributes also support the basis for role-based access controls. This real-time situational awareness capability is one of the most important features that will improve response and reaction time, as it removes the necessity to read and process every single message that flows through the system at a given time, or to manually trigger reports or searches to get situational awareness.

[[downstreamCollection]] Downstream Collection

The DIMS system should support collection of event data, network flow data, and other security related event data from upstream collector nodes at edge networks that participate in such collection. (This requirement covers the central, or downstream, collection servers.)

[[bdsUserStory1]] BDS User Story 1

“As {an investigator, analyst} I want to be able to preserve the results of searches, and in some cases the data that was identified while searching, in order to have copies that are subject to expiration and purging from the system. Some investigations may take many months, which could bump up against the data retention period (approximately 12 months, at present).”

[[bdsUserStory2]] BDS User Story 2

“As {a security operator, investigator, analyst, CISO} I want to be able to define multiple sets of attributes that the system can then use to inform me about when new data is seen that matches those attributes. Attributes can include anything that might be seen in indicators of compromise, observables, or alerts. (The most basic being IP addresses and/or CIDR blocks, domain names, MD5 or other cryptographic hash values, file names, Registry key settings, etc.)”

3.2.2 Dashboard Web Application (DWA) CSCI

The Dashboard Web Application, also referred to as the DIMS Dashboard, provides a multi-faceted user interface and data visualization and analytic tools to integrate data from many sources and facilitate trusted information sharing. The following subsections contain the user stories which describe the Dashboard Web Application requirements.

[[dwaUserStory1]] DWA User Story 1

“As {an investigator, analyst} I want to be able to keep track of cases and campaigns (i.e., groups of related incidents). I want the system to inform me, if I so chose, of any time new data that is determined to be associated with the sets I am tracking comes into the system. For example, if I log in and open a case, I can easily tell which data has been entered into the case since the last time viewed the case. This allows me to stay on top of new evidence or activity that I am investigating.”

[[dwaUserStory2]] DWA User Story 2

“As {a security operator, investigator} I want to be told when an email thread or received set of indicators includes systems that I am responsible for securing, ideally pointing out to me those hosts that are involved without requiring that I read the entire thread, extract attachments, write scripts to parse and search data, etc. I want to be given a list of those records that are important, in a format that I can submit directly to query interfaces without having to write scripts to parse and process."

[[dwaUserStory3]] DWA User Story 3

"As an {analyst, investigator, security operator}, I would like to be able to get context about 'external' hosts that includes what kind of malicious activity has been observed, by whom, starting and ending when, have they been involved in precious incidents I have dealt with, etc. This view could combine a timeline aspect (first seen to last seen time ranges along the X axis), for one or more sources of threat intelligence (discrete items along a non-linear Y axis) with some method of mapping to these external hosts (grouping into AS, etc.). The objective is to quickly associate context about threats within observed flows or logged events."

[[dwaUserStory4]] DWA User Story 4

"As an {analyst, investigator, security operator}, I would like to be able to step through large volumes of output records in a manner that reduces the set of remaining items as quickly as possible. I would like to see related entries visually identified as being part of a common set, and have the ability to select one representative entry, tag it, categorize it as being benign or malicious, then filtering all of the related records out so as to focus on categorizing the remaining records. If the system can remember the tags and automatically apply them when similar records are seen in the future, it will be easier to identify new unknown records that require analytic scrutiny."

[[dwaUserStory5]] DWA User Story 5

"As an {analyst, security operator}, I would like to have links to detailed analyses and reports that are available in public sources when a query I have made results in identifying known malware or malicious actors. This way I can more quickly come up to speed on what is (or is not) known about the threat behind the indicators or observables I am dealing with."

[[dwaUserStory6]] DWA User Story 6

"As a {system administrator, security operator, network operator}, I would like to have links to Course of Action steps related to the threats that I identify using the DIMS system. This allows me to not only inform owners or compromised assets that have been identified by the system, but to also give them information about what they need to do, in what order they should take steps, and when/how to preserve evidence in the event that there is criminal investigation ongoing."

[[dwaUserStory7]] DWA User Story 7

"As an {analyst, security operator, investigator, network operator, system administrator}, I would like to be able to have access to DIMS functions via an intuitive web user interface."

[[dwaUserStory8]] DWA User Story 8

"As a system administratory, I want the DIMS Dashboard to report information upon system startup and at periodic intervals that indicate operational status."

[[dwaUserStory9]] DWA User Story 9

"As a potential new user of the system, I would like to be able to request to be invited to join." Without already having an account on the system, a potential new member should be able to trigger an email to a specific administrators email list with their name, email address, and other necessary identifying information sufficient to be nominated to a trust group.

3.2.3 Data Integration and User Tools (DIUT) CSCI

The following sections describe the requirements for the Data Integration and User Tools (DIUT) CSCI.

[[incidentTracking]] Incident/Campaign Tracking

The DIMS system must be able to keep track of multiple incidents, campaigns, sector-specific threat activity, or other ad-hoc groupings of security information as desired by DIMS users. For example, an analyst may wish to track ZeroAccess trojan activity, CryptoLocker extortion attempts, Zeus or Citadel ACH fraud attempts, etc., possibly over time periods measured in years. Each user may wish to label these associated sets with their own labels, or may want to use a system-wide naming scheme that conforms to an ontology that is more rigorously defined. These sets should be easily shared with other users.

[[knowledgeAcquisition]] Knowledge Acquisition

The DIMS system should support knowledge acquisition by allowing the user to be told, on login and when they focus on a particular incident or campaign, what new information has been obtained from other users of the system (or the system itself through automated detection and reporting) since the last time the user was reviewing the incident or campaign. Collaboration works best when team members learn from each other, and the asynchronous nature of a multi-user system is such that determining the delta in knowledge since an earlier point in time is difficult to achieve.

[[aggregateSummary]] Summarize Aggregate Data

The DIMS system should summarize any/all aggregate data that any user is presented with sufficient context to quickly understand the data. This includes (but is not limited to): Start and end date and time; Total number of systems within the "friend" population, and how they break down across participants; Total number of systems outside of the "friend" population, and how they break down by country/AS/IP address(es); Total number of systems from the "not-friend" population that are known to be malicious (a.k.a., "foe"), broken down by country/AS/IP address(es). When the number of IP addresses exceeds a certain threshold, they are summarized in aggregate, with a mechanism to dig down if the user so chooses. Similarly, context about what quantity and quality of malicious activity that is known about the "foe" population should also be available for easy access (presented if short, or drill-down provided it too voluminous). This amount and level of detail provides an overall "situational awareness" or scoping of large volumes of security event data. (The mechanism for such multi-level tabular reports is known as "break" or "step" reports).

Note: You can find an example of a break report in example **1.37 Grouping rows by a given key (itertools.groupby)** in 30 Python Language Features and Tricks You May Not Know About, by Sahand Saba, May 19, 2014. The example from that page is included below.

The DIMS Test Report itself is produced using a break report that categorizes output broken down by Test levels.

```
>>> data.sort(key=itemgetter(-1))
>>> for value, group in itertools.groupby(data, lambda r: r[-1]):
...     print '-----------'
...     print 'Group: ' + value
...     print_data(group)
...
-----------
Group: hard
young               myope                   yes                     normal                  hard
young               hypermetrope            yes                     normal                  hard
pre-presbyopic      myope                   yes                     normal                  hard
presbyopic          myope                   yes                     normal                  hard
-----------
Group: none
young               myope                   no                      reduced                 none
young               myope                   yes                     reduced                 none
young               hypermetrope            no                      reduced                 none
young               hypermetrope            yes                     reduced                 none
pre-presbyopic      myope                   no                      reduced                 none
pre-presbyopic      myope                   yes                     reduced                 none
pre-presbyopic      hypermetrope            no                      reduced                 none
pre-presbyopic      hypermetrope            yes                     reduced                 none
pre-presbyopic      hypermetrope            yes                     normal                  none
presbyopic          myope                   no                      reduced                 none
presbyopic          myope                   no                      normal                  none
presbyopic          myope                   yes                     reduced                 none
presbyopic          hypermetrope            no                      reduced                 none
presbyopic          hypermetrope            yes                     reduced                 none
presbyopic          hypermetrope            yes                     normal                  none
-----------
Group: soft
young               myope                   no                      normal                  soft
young               hypermetrope            no                      normal                  soft
pre-presbyopic      myope                   no                      normal                  soft
pre-presbyopic      hypermetrope            no                      normal                  soft
presbyopic          hypermetrope            no                      normal                  soft
```

[[upstreamCollection]] Upstream Collection

The DIMS system should support collection of event data, network flow data, and other security related event data from upstream collector nodes at edge networks that participate in such collection. (This requirement covers the upstream collector nodes.)

[[diutUserStory1]] DIUT User Story 1

"As an investigator, I would like to be able to timestamp files I create (i.e., calculate multiple different cryptographic hashes of the contents of files to validate their integrity, associate a timestamp from a trusted time source, then cryptographically sign the result with a private key). This allows validation of the existence of a file at a point in time, who produced the file, and maintenance of a form of "chain of custody" of the contents of the file. To ensure privacy as well as integrity and provenance, the file would first be encrypted (or both cleartext and encrypted files included in the timestamping operation)."

[[diutUserStory2]] DIUT User Story 2

"As a system administrator, I would like to have a picture of the operational state of all of the system components that make up DIMS (and related underlying SIEM, etc.) This will allow me to quickly diagnose outages in dependent sub-systems that cause the system as a whole to not function as expected. The less time that it takes me to diagnose the trouble and remediate, the better."

[[diutUserStory3]] DIUT User Story 3

"As a system administrator, I would like to be able to update or reconfigure DIMS subsystem components from a central location (rather than having to log in to each system and copy/edit files by hand). I would like to be assured that those changes are applied uniformly across all subsystem components, and that I have a mechanism to back out to a previous running state if need be to maintain uptime."

[[diutUserStory4]] DIUT User Story 4

"As a {system administrator, security operator}, I would like to know that the DIMS system components are being monitored for attempted access by any of the same malicious actors who are seen to be threatening my constituent users. It is only natural to assume that an attack on any participant site could lead to discovery of the security monitoring system and for that system to be attacked as well, so the system should be monitoring itself using the same cross-organizational correlation features as are used internally."

[[diutUserStory5]] DIUT User Story 5

"As a system administrator, I would like to be able to deal with a breach of the security system in a tactical way. If a user is found to have had a compromise of their account, all access to that user should be disabled uniformly across all system components via the single-signon authentication subsystem. All cryptographic keys should also be revoked. Once the user has been informed and the computer systems they use cleaned, all cryptographic keys, certificates, and password should be updated and re-issued."

[[diutUserStory6]] DIUT User Story 6

"As a {system administrator, security operator}, I would like to be able to link indicators and observables that come in at the network level (e.g., IP addresses, domain names, URLs) to observables at the host level (e.g., Registry Keys and values, file names, cryptographic hashes of files) and search for those observables to confirm or refute assertions that computers under my authority have been compromised. If I get confirmation, I would then like to preserve evidence and maintain chain of custody for that evidence as easily and quickly as possible."

[[diutUserStory7]] DIUT User Story 7

"As an {analyst, security operator} I would like to be able to start an analysis and annotate data files as I go through the analysis process, trying to derive meaning from what I am seeing in the data, and being able to (at any time seems appropriate) create a reference to the current data set(s) and my view of them so I can pass this reference identifier to another analyst, a CISO, or an investigator, to allow them to take a look at what I am seeing and provide their input. For example, if someone reports a DoS attack directed at SLTT government, and my analysis confirms that such an act can be seen in the PRISEM population, I would like to provide my observations to someone to help investigate targeting, etc., in order to develop a better picture of what is happening. If the result is a determination that a SITREP should be developed and information passed along to federal law enforcement, the updated annotated body of data can then be assembled into a SITREP (using a 'break' or 'step' reporting format, including both cleartext and anonymized versions for sharing with outside groups) and passed along with little added effort."

[[diutUserStory8]] DIUT User Story 8

"As a user of the system, I would like to see the status of any asynchronous queries or report generation requests I have made. It is reasonable for a search through the entire history of billions of events to take some time to complete, but I would like to be able to tell approximately how long I will have to wait. Ideally, the system would keep track of previous requests, the time span and complexity of filtering applied, and to provide a time estimate when a new query is being formulated so as to guide me in deciding what I really need to ask for to get an answer in the time frame I am faced with at the moment."

3.2.4 Vertical/Lateral Information Sharing (VLIS) CSCI

The following sections describe the requirements for the Vertical and Lateral Information Sharing (VLIS) CSCI.

[[structuredInput]] Structured data input

The DIMS system must have the ability to process structured data that is entered into the system in one of several ways: (1) attached to email messages being sent to the Ops-Trust portal (optionally as encrypted attachments); (2) via CIF feed, TAXII, AMQP message bus, or other asynchronous automated mechanism; (3) as uploaded from a user's workstation via the DIMS dashboard client; (4) via the Tupelo client or other command line mechanism.

[[assetIdentification]] Asset Identification

The DIMS system must be able to detect when IP addresses or domain names associated with a given set of CIDR blocks or top-level domains are involved, and to trigger one or more workflow processes. This could be to send an alert to a user when some entity they are watching is found in a communication, generate a scheduled report, or trigger some other asynchronous event. It may be to initiate a search of available data so the results can be ready for a user to view when they receive the alert, rather than requiring that they initiate a search at that time and have to wait for the results.

[[vlisUserStory1]] VLIS User Story 1

"As a user of the DIMS system, I would like the ability to (at any point in time during analysis of an incident or while viewing the situation associated with threats across the user population) produce an anonymized version of the output I am looking at so as to be able to share it with outside entities. The system should anonymize and filter the data according to the policies set by the entities that provided the underlying data, and I should be able to determine the policy for sharing of information (by clearly seeing its tagged TLP sensitivity level). Reports should similarly be tagged appropriately with TLP for the sensitivity level of the aggregate document."

3.3 Adaptation requirements

The DIMS system will be designed so as to use a set of operational parameters specific to the deployment and user, in order for the system to function as a normal internet-accessible service using TCP/IP and DNS. These attributes include (but are not limited to):

- Top level domain name (e.g., `prisem.washington.edu`, or `test.prisem.washington.edu`)
- External IP network address block in CIDR notation (e.g., `140.142.29.0/24`)
- External IP address of the primary service access point providing web portal, dashboard web application, VPN server, etc. (e.g., `140.142.29.101`)

- Description of organization for branding (e.g., "Public Regional Information Security Event Management")
- Logo for branding
- Internal IP network address block in CIDR notation (e.g., `10.1.0.0/16`)
- Internal NAT gateway address (e.g., `10.1.0.1`)

These parameters will be stored in a configuration database that will be used to configure the system services, network interfaces, brand the documentation, customize the appearance of the web application user interface, etc. for the specific deployment.

Note: As shown in the domain name examples above, an extra level of domain name system hierarchy may be used to differentiate multiple deployments of DIMS within an organization for the purposes of separating development from test/evaluation from "production", so the following naming scheme may be used (where the same host name `webapp` or `vpn` may exist uniquely in each of the separate domain name spaces where `*` is shown):

Table 3.1: Segmented domains

System purpose	Stability	Top level domain
Development	Unstable	`*.dev.prisem.washington.edu`
DevTest	Changes within sprint	`*.test.prisem.washington.edu`
Evaluation/Demos	Changes on sprint cycle	`*.demo.prisem.washington.edu`
Production	Changed on release cycle	`*.prisem.washington.edu`

3.4 Security and privacy requirements

3.4.1 [[networkAccessControls]] Network Access Controls

Remote users need to access DIMS components in order to use the system. Direct internet access is necessary for a limited subset of DIMS components, while the remainder are to be restricted to indirect access through the Dashboard Web Application and ops-trust portal front end, or by restricted access through a Virtual Private Network (VPN) connection.

1. The features described in *Dashboard Web Application (DWA) CSCI* are to be accessible from the internet from a limited set of network ports.
2. The features described in *Backend Data Stores (BDS) CSCI* are primarily only accessible to other CSCI components on a restricted network and have little or no direct user interface, while some features described in Section *Data Integration and User Tools (DIUT) CSCI* and Section *Vertical/Lateral Information Sharing (VLIS) CSCI* may have user Command Line Interfaces (CLIs) or Application Programming Interfaces (APIs) accessible only when the user is connected by VPN, or through SSH tunneling.

Ideally, all internet access to user interfaces (either graphical or command line) will be through a single IP address via direct connection, through a proxy connection, or to firewalled hosts via Network Address Translation (NAT) and/or Port Forwarding (a.k.a., Destination NAT or DNAT). This is to reduce the number of internet routable IP addresses and DNS names for a DIMS deployment to just one, as well as to simplify access control and access monitoring.

3.4.2 [[accountAccessControls]] Account Access Controls

All DIMS component services should have access controls allowing only authorized users access. The primary mechanism for doing this is the use of a *Single Sign-On* (SSO) system and authentication service.

3.4.3 [[secondFactorAuth]] Second-factor authentication

The DIMS system should support the use of *two factor authentication*. The ops-trust portal code base supports:

1. Time-base One Time Password (TOTP)
2. HMAC-based One Time Password (HOTP)
3. Static single use codes (a list of codes you can use to authenticate if all else fails)

The principle supported application for two-factor authentication is Google Authenticator.

3.4.4 [[accountSuspension]] Account suspension

When an account is suspected of being compromised, all access for that user should be suspended in a manner that is non-destructive (i.e., access is removed, but no credentials or account contents are deleted.) This allows an account to be toggled off while an investigation takes place, and back on again once the account has been deemed secure. Use of a *single-signon* (SSO) mechanism can facilitate this, but additional mechanisms to remove access must also be taken into consideration. For example, SSL client certificates (e.g., those used with OpenVPN).

3.4.5 [[keyRegeneration]] Key Regeneration and Replacement

Cryptographic keys are used for secure access to many DIMS components, including SSH public/private key pairs, and SSL client certificates for OpenVPN access. Certificates should be generated for the user automatically as a workflow process step performed by the system when a new account is activated in the ops-trust portal.

There should also be a way for user certificates to be regenerated (e.g., when someone's laptop is compromised by malware, or is lost/stolen), and a way to selectively (or wholesale) regenerate certificates for any/all users (e.g., when a DIMS system component suffers a breach.)

These security mechanisms allow restoration of a secure system with the least amount of time/energy as possible.

3.5 Design and implementation constraints

3.5.1 [[automatedProvisioning]] Automated Provisioning

The DIMS server components must be provisioned, configured, and administered from a single central location and pushed to servers in an automated fashion. Manual configuration and patching of hosts takes too much expert system administration knowledge, incurs too much system administration overhead, and takes too long to recover from outages or system upgrades. The DIMS team will be administering multiple instances of the DIMS system (for development, alpha testing, beta testing, a "production" PRISEM instance for in-field test and evaluation, and potentially 3-5 more instances at other regions (see the Stakeholders section). It will be impossible to manually manage that many deployments with current staffing levels.

3.5.2 [[agileDevelopment]] Agile development

The system will be built using an Agile coding methodology, responding to user feedback as quickly as possible to ensure maximum usability and scalability. The desired release cycle (length of a "sprint") is 2-3 weeks.

3.5.3 [[continuousIntegration]] Continuous Integration & Delivery

The systems running DIMS software must support continuous integration of code releases, updating runtime executables, stopping and starting service daemons, etc., in a controlled, predictable, and repeatable manner. Runtime components must identify the source code release from which they were built (e.g., “v2.1.20” or “v2.1-56-g55a5d1”) in order to track bugs and features across multiple deployments with a regular release cycle.

3.5.4 [[leverageOpenSource]] Leveraging open source components

As much as possible, DIMS will be built through the (re)use of open source components used by other projects that are being integrated into the DIMS framework. For example, the Collective Intelligence Framework (CIF) v2 and the Mozilla Defense Platform (MozDef) both employ the ELK stack and RabbitMQ in their demonstration implementations, and the original PRISEM distributed data processing tools also used RabbitMQ. Rather than have two separate instances of Elasticsearch running in virtual machines or containers for MozDef and CIF, and two separate instances of RabbitMQ in virtual machines or containers for PRISEM tools and MozDef, a common Elasticsearch cluster and RabbitMQ cluster would be set up and shared with these (and any other open source tools added later).

3.6 Other requirements

3.6.1 [[exportControl]] Export control

The software produced under this contract is subject to export control restrictions on encryption components. Any software libraries, or encryption keys, *must* be acquired or produced by the end user implementing DIMS, *not* distributed as part of the DIMS code base. The plan is to release software components with instructions on how to acquire and install the necessary cryptographic elements *before* beginning the installation process.

3.7 Packaging requirements

3.7.1 [[noEncryption]] No included cryptographic elements

Per Section *[[exportControl]] Export control*, all software packaged for release *must* have checks to confirm that cryptographic libraries and/or encrypt keys are *not present* in the packaged source or delivered system component(s).

3.7.2 [[openSourceRelease]] Open source release

All DIMS source code will be released through GitHub at https://github.com/uw-dims under the license found in Section *License*. All documentation will be released at GitHub and/or Read the Docs under the same license.

CHAPTER 4

Notes

This document is structured on MIL-STD-498, described at A forgotten military standard that saves weeks of work (by providing free project management templates), by Kristof Kovacs. Specifically, this document is modelled on SSS.html.

4.1 Glossary of Terms

Agile A programming methodology based on short cycles of feature-specific changes and rapid delivery, as opposed to the "Waterfall" model of system development with long requirements definition, specification, design, build, test, acceptance, delivery sequences of steps.

Botnets System The name given to the re-implementation of *Einstein 1* technology. See http://web.archive.org/web/20131115180654/http://www.botnets.org/

cron A Unix/Linux service daemon that is responsible for running background tasks on a scheduled basis.

CIFglue "Simple rails app to quickly add indicators to the Collective Intelligence Framework"

Cryptographic Hash, Cryptographic Hashing Algorithm A mathematical method of uniquely representing a stream of bits with a fixed-length numeric value in a numeric space sufficiently large so as to be infeasible to predictably generate the same hash value for two different files. (Used as an integrity checking mechanism). Commonly used algorithms are MD5, SHA1, SHA224, SHA256, RIPEMD-128. (See also http://en.wikipedia.org/wiki/Cryptographic_hash_function).

CSCI An aggregation of software that satisfies an end use function and is designated for separate configuration management by the acquirer. CSCIs are selected based on tradeoffs among software function, size, host or target computers, developer, support concept, plans for reuse, criticality, interface considerations, need to be separately documented and controlled, and other factors.

Einstein 1 A network flow based behavioral and watchlist based detection system developed by University of Michigan and Merit Networks, Inc. for use by US-CERT. The re-implementation is known as the *Botnets System*.

Fusion Center Entities created by DHS to integrate federal law enforcement and intelligence resources with state and local law enforcement for greater collaboration and information sharing across levels of SLTT governments.

Git A source code version management system in widespread use.

MUTEX Mutual Exclusion (object or lock, used to synchronize execution of independent threads or processes that must share a common resource in an exclusive manner, or to ensure only one copy of a program is running at a time)

NetFlow Record format developed by Cisco for logging and storing Network Flow information (see also SiLKTools).

NoSQL The term for database that does not use the typical table-based relational schema as Relational Database Management Systems (RDBMS)

Ops-Trust (ops-t) Operational Security Trust organization (see http://ops-trust.net/)

Port forwarding A mechanism used by NAT firewalls to forward a connection by port number to a host behind the NAT firewall. Also known as Destination NAT, or "DNAT". (See NAT, DNAT.)

Redis A "NoSQL" database system used to store files in a key/value pair model via a RESTful HTTP/HTTPS interface.

SiLKTools A network flow logging and archiving format and tool set developed by Carnegie Mellon's Software Engineering Institute (in support of CERT/CC).

Team Cymru (Pronounced "COME-ree") – "Team Cymru Research NFP is a specialized Internet security research firm and 501(c)3 non-profit dedicated to making the Internet more secure. Team Cymru helps organizations identify and eradicate problems in their networks, providing insight that improves lives."

Tupelo A host-based forensic system (client and server) developed at the University of Washington, based on the Honeynet Project "Manuka" system.

4.2 List of Acronyms

AAA Authentication, Authorization, and Accounting

AMQP Advanced Message Queuing Protocol

API Application Programming Interface

AS Autonomous System

ASN Autonomous System Number

CI Critical Infrastructure

CIDR Classless Internet Domain Routing

CIF Collective Intelligence Framework

CIP Critical Infrastructure Protection

CISO Chief Information and Security Officer

CLI Command Line Interface

COA Course of Action (steps to Respond and Recover)

CONOPS Concept of Operations

CRADA Cooperative Research and Development Agreement

CSIRT Computer Security Incident Response Team

CSV Comma-separated Value (a semi-structured file format)

DNAT Destination NAT (see NAT and "port forwarding")

DIMS Distributed Incident Management System

DNS Domain Name System

DoS Denial of Service

DDoS Distributed Denial of Service

EO Executive Order

GZIP Gnu ZIP (file compression program)

HSPD Homeland Security Presidential Directive

ICT Information and Communication Technology

IOC Indicators of Compromise

IP Internet Protocol (TCP and UDP are examples of Internet Protocols)

IRC Internet Relay Chat (an instant messaging system)

JSON JavaScript Object Notation

MAPP Microsoft Active Protections Program

MNS Mission Needs Statement

NCFTA National Cyber-Forensics & Training Alliance

NAT Network Address Translation

NTP Network Time Protocol (a service exploited to perform reflected/amplified DDoS attacks by spoofing the source address of requests, where the much larger responses flood the victim)

OODA Observe, Orient, Decide, and Act (also known as the "Boyd Cycle")

PPD Presidential Policy Directive

PRISEM Public Regional Information Security Event Management

RBAC Role Based Access Control

RESTful Representational State Transfer web service API

RPC Remote Procedure Call

SCADA Supervisory Control and Data Acquisition

SIEM Security Information Event Management (sometimes referred to as Security Event Information Management, Security Event Monitoring, causing some to pronounce it as "sim-sem".)

SLTT State, Local, Territorial, and Tribal (classification of non-federal government entities)

SOC Security Operations Center

SoD Security on Demand (PRISEM project support vendor)

SSH Secure Shell

STIX Structure Threat Information Expression. A standard for information exchange developed by MITRE in support of DHS US-CERT.

TAXII Trusted Automated Exchange of Indicator Information

TCP Transmission Control Protocol (one of the Internet Protocols)

TLP Traffic Light Protocol

TTP Tools, Tactics, and Procedures

UC Use Case

UDP Unreliable Datagram Protocol (one of the Internet Protocols)

WCX Western Cyber Exchange

CHAPTER 5

License

Section author: Dave Dittrich (@davedittrich) <dittrich @ u.washington.edu>

```
Berkeley Three Clause License
=============================

Copyright (c) 2014 - 2016 University of Washington. All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this
list of conditions and the following disclaimer.

2. Redistributions in binary form must reproduce the above copyright notice,
this list of conditions and the following disclaimer in the documentation
and/or other materials provided with the distribution.

3. Neither the name of the copyright holder nor the names of its contributors
may be used to endorse or promote products derived from this software without
specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND
ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED
WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE
FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL
DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR
SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER
CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY,
OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE
OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
```

Index