FINAL TECHNICAL REPORT

Energy Sector Security through a System for Intelligent,
Learning Network Configuration Monitoring and Management

Essence tm

National Rural Electric Cooperative Association
Business and Technology Strategies

NRECA
America's Electric Cooperatives

Federal Agency: United States Department of Energy

Identifying Number: DE-OE0000684

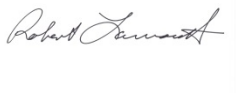
Contents

FINAL TECHNICAL REPORT

1. DATA ELEMENTS

Federal Agency:	United States Department of Energy
Identifying Number:	DE-OE0000684
Project Title:	Energy Sector Security through a System for Intelligent, Learning Network Configuration Monitoring and Management ("Essence")
Project Manager: Email: Phone:	Robert Larmouth robert.larmouth-contractor@nreca.coop 603-930-9199
Principal Investigator: Email: Phone:	Dr. Craig Miller craig.miller@nreca.coop 703-626-9683
Submission Date:	May 31, 2017
DUNS Number:	045497427
Recipient Organization: Address:	National Rural Electric Cooperative Association 4301 Wilson Boulevard Arlington, VA 22203-1860
Grant Period:	October 1, 2013 through December 31, 2016
Reporting Period End Date: Report Term or Frequency:	December 31, 2016 Final

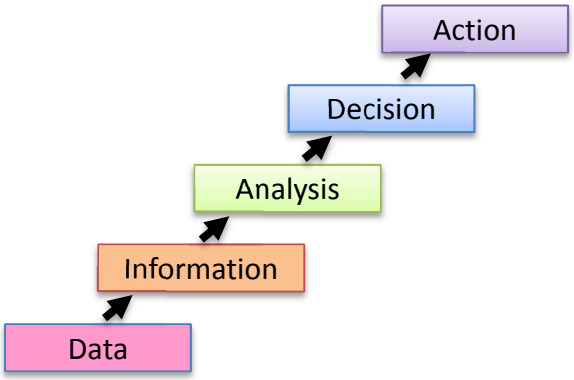
Signature of Submitting Official:



Robert Larmouth

2. EXECUTIVE SUMMARY

The project was conceived and executed with the overarching objective to provide cost effective tools to cooperative utilities that enabled them to quickly detect, characterize and take remediative action against cyber attacks. The architecture of the solution was derived from the following five-layer abstraction model:



From this model, a working prototype system was developed through a two-cycle iteration of the following project elements:

1. High level design
2. Component design
3. Network design
4. Lab testing
5. Field testing
6. Commercialization plan

Design and development work proceeded largely on schedule and the team was able to successfully test the first prototype at three (3) North Carolina cooperative utilities May and June of 2015. Feedback from the testing enabled the team to enhance the system features such as rules implementation and more intuitive user interface. At the end of cycle 2, the system was ready for the commercialization task, although this phase was limited in scope to a plan, not an actual commercial product.

The goal during the commercialization task was to demonstrate the process to integrate Essence with the Managed Security Services (SEDC MSS) offering of Southeast Data Cooperative (SEDC). SEDC MSS, through its AlienVault product, covers the entire electric utility's network and includes protection for network intrusion detection, host intrusion detection, behavioral analysis, log collection, and network traffic analysis. Integrating Essence with SEDC MSS:

- Provides a single interface to present information from Essence and SEDC MSS,
- Conducts cross-correlation of information collected by Essence and SEDC MSS,

- Decreases time required to deploy Essence appliance (if SEDC MSS or AlienVault is already in service at an electric utility)

The results of this work can be looked at as a proof-of-concept. Any ultimate product offering by a provider such as SEDC will benefit from this integration demonstration. The build and test process is summarized in section 4.

In a testament to its success, the Essence system has since been leveraged in three subsequent projects:

1. GridState – rapid anomaly detection for critical infrastructure (DARPA RADICS)
2. RC3 – assessment tools and training (DOE)
3. React – commercialized products (DOE CEDS)

3. ACCOMPLISHMENTS

3a Project Team

The project team, with NRECA as the lead, consisted of the following partners:

- Cigital, Inc (recently acquired by Synopsys)
- Carnegie Mellon University
- Pacific Northwest National Laboratory (through September 2015)
- Honeywell, Inc (through February 2015)
- Cooperatives

3b Results of Major goals of the project

The original high-level goals of this project included development of the following capabilities:

- GOAL - A capability to establish, maintain and monitor an alignment between OT security policy and network configuration and settings.

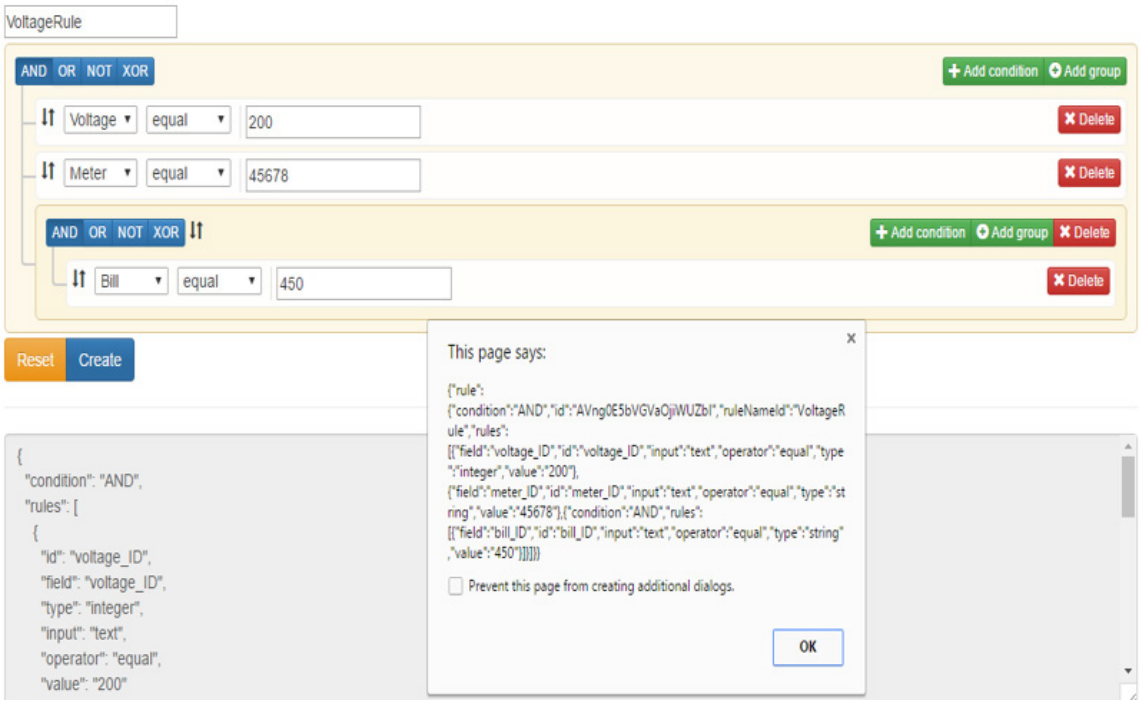
  This goal was fully met.

  "Policy", as it is used in this goal, refers to the authority of particular users (identified by address) to communicate with and control particular systems also identified by address. Essence enables the encoding of these policies in rules allowing white listing or black listing. Rules can specify that communications are prohibited between end points or communications with an end point are only allowed from specific other end points. Because Essence considers more than packet structure, it can extract payload information which can also be used in white and black lists. For example, a rule can be written that only messages using a particular engineering protocol be sent from an end point address or received by an end point.

- GOAL - A capability to detect and prevent potentially malicious traffic flows within an electric utility's operational network, by creating protocol-specific (e.g., specific to MultiSpeak™, DNP3), semantically rich, and context-aware filtering rules to identify disallowed or anomalous traffic patterns.

  This goal was fully met

  Essence has access to all aspects of a packet's structure and payload. Rules can be written against any of these or any combination using a rules interface. An example is shown here:

Rules composition interface


  Ongoing work in related projects will add a capability to create rules on calculated values.

- GOAL - A capability to enable an electric utility to define and enforce its operational network security policies with fewer IT staff members and less reliance on significant internal security expertise.

  This goal was fully met.

  Three features of Essence address this goal:

  (1) A rules creation engine – shown above
  (2) Rules derived from machine learning (discussed below)
  (3) The ability to import rules from other systems and operators.

As shown in the illustration of the rules interface, the rules are encoded and stored as JSON (Java Script Object Notation) which can be moved from system to system, or edited directly by a skilled user.

- GOAL - A capability to align an electric utility's operational network security management with the broader trends of software-defined networking, virtualization, and the ongoing migration of utility IT and operational systems into the cloud environment (where they can be provided as a managed service). This alignment is particularly important for smaller electric utilities that have limited IT staffs and capabilities.

  This goal was 90% met.

  A capability was developed to take action using software-defined networking (SDN) to isolate a communication from a suspect source. The user was offered the option to continue using data from the source, to sever communications from the source, or to embargo data from the source for forensic analysis or use after further analysis. This was implemented using the OpenDaylight SDN specification. Unfortunately, we did not find the then-current implementation of OpenDaylight to be sufficiently robust and complete to proceed with commercial development.

  The work in the project demonstrated that, at a future date when SDN is more mature and widely deployed, networks can be reconfigured in response to cyber-attacks.

- GOAL - A capability to simplify functions such as security reporting and compliance as they relate to an electric utility's operational network assets and traffic flows.

  This goal was 100% met.

  Essence provides a very efficient method for mapping a utility's network, based on Bro and detecting and reporting anomalies in reference to the map.

The task chart below reflects the successful completion of work per the original project plan.

CYCLE 1: PROTOTYPE DEVELOPMENT					
Task No.	**Task**	**START (week)**	**END (week)**	**DURATION (weeks)**	**Status as of Dec 31, '16**
1	Project Management and Planning	0	130	130	100%
2	High-Level Design	2	11	9	100%
3	Component Design and Implementation	12	49	37	100%
4	Network System Design	12	49	37	100%
5	Laboratory Testing	48	66	18	100%
6	Field and User Testing	56	67	11	100%

CYCLE 2: REFINEMENT AND COMMERCIALIZATION					
2	Refine High-Level Design	68	88	20	100%
3	Refine Component Design And Implement	68	88	20	100%
4	Refine System Design and Implement Changes	88	102	14	100%
5	Laboratory Testing	102	106	4	100%
6	Refined Field and User Testing	106	114	8	100%
7	Commercialization	84	122	38	100%

There were six (6) go/no-go milestones successfully achieved during the project:

When (week)	Milestone	Go/No-go Decision?
11	M3: Prototype Design Complete and meets objectives	Complete
67	M6: Utilities successfully recruited and ready for field testing	Complete
67	M6: Lab testing completed and meets objectives of first design	Complete
79	M7: Field testing complete and results meet objectives	Complete
88	M8: Cycle 2 design complete and meets objectives	Complete
114	M11: Testing of Cycle 2 design complete and meets objectives to proceed to field testing	Complete

4. PROJECT ACTIVITY SUMMARY

Accomplishments of Task 1 – Project Management

The project was managed as planned. NRECA used a Team/Charter method in which the project work was divided among teams, each of which operated with a relatively high-degree of autonomy in regards to approach. A team lead was designated. The lead staffed the tasks and the team developed a "charter" which described the work to be done and how it would be accomplished. The Project Manager and the Principal Investigator reviewed the charters and adjust them with the teams to provide alignment. The charters were previously submitted.

Accomplishments of Task 2 – High Level Design

The high-level design was implemented as originally conceived, around the five level design discussed previously. The design document was previously submitted.

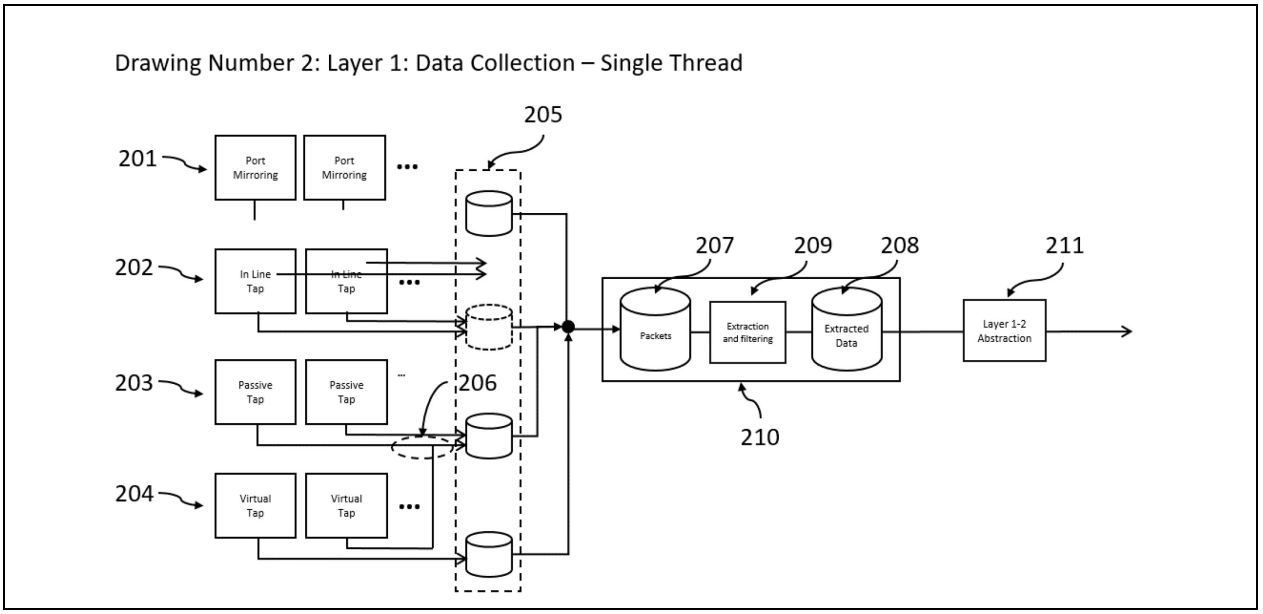
Accomplishments of Task 3 – Component Design

The Component Design addressed the implementation within each of the layers in the abstraction model shown in the Executive Summary.

Layer 1: Data

In the data layer, NRECA developed multiple capture technologies. These are described here and within a section of the Essence patent application.

Drawing – Layer 1: Data Collection



Drawing Number two shows the general structure of the first abstraction layer of Essence. This is the data collection layer, which captures data from multiple sources and passes the data to Layer 2. The objective of this layer is to capture as much of the utility's telemetry, controls signals, and operations (as opposed to business) communications as possible and transmit this to layer 2 without loss and with minimal latency. Data may be collected at one or more points of diverse kinds, described below. The data are combined into a single connection to the Layer 2 so new data collection points can be added without re-engineering the interface between Layer 1 and Layer 2.

201 – Data can be collected through port mirroring. Port mirroring is a feature built into ethernet switches and routers and some grid devices such as SCADA systems. On the common Cisco routers, the mirror port is termed a SPAN port for "switched port analyzer". When port mirroring is turned on, a copy of traffic from any one port or all ports is copied to the mirror ports where it is communicated to another address for storage or analysis. An Essence installation may include collection from zero, one, or many ports.

202 – An inline tap is effectively a dedicated port mirroring device. Functionally, this kind of network tap is identical to a mirrored port, except that it is not part of a switch or a router. Again, an Essence installation may have zero, one, or many dedicated inline taps.

203 – Passive taps differ from inline taps in not providing an electrical connection from the standard mirror port; tap duplicate lines include a feature (termed a data diode) to prevent signals from flowing from the duplicate line back into the device. This prevents the Essence line from being used to attack the tap. This method is preferred to methods which do not include such isolation. There may be zero, one or many passive taps.

204 - A virtual tap provides the functional equivalent of a physical tap (elements 201, 202, and 203) but is implemented purely in software and collects data from communications between virtual machines. Virtual machines are images of two or more computers operating on a single computer or in the cloud. The virtual computers operate like standalone computers and communicate with each other by the same means that physical computers do (e.g. TCP/IP), except that the communications do not pass over physical network cables. A virtual tap is an implementation in software to replicate such communications for input to Essence Layer 2. There may be zero, one, or many virtual taps. There must, however, be at least one of (201,202,203, or 204). That is, and Essence system requires some data input.

205 – All data from collections points (201,202,203,204) are first passed to one local data store dedicated to collection of such data. This store provides a buffering function to allow data collected for storage to be forwarded over a limited period before it is forwarded for processing and transmission to Layer 2.

206 – One local data store can receive information from one or more collection points including collection points of different types.

207 – The Packet Collection Database is a large, fast data store that receives data from the local data stores. The data are stored as packets in the order received. This flat structure, as opposed to a relational structure, provides for high speed as it is critical that no data be lost. It is possible that data related to a particular measurement or control instruction may be received from more than one data collection point. Based on the information in the packet, these may be precisely synchronous or they may be slightly out of phase due to differences in the latency of the two collection paths. A path is a collection point plus its associated local data store and the connections (physical and /o r virtual) used to transmit the data to the Collection Database. In either case, it is necessary to remove duplicates but this function is not performed in Layer 1 (Data Layer). The emphasis in the Layer 1 is on speed to prevent the risk of data loss, so as little processing of the data as possible is performed.

208 – Extracted Data Database contains information extracted from the Packet Collection. While, as noted, as little processing of the packet information as possible is done in Layer 1, the decomposition of packets is essential. Packets consist of two basic types of data – packet information related to the construction of the actual packet and payload – the content of the packet. The packet information for Internet Protocol Version 4 is structured as follows. Essence can work with IP Version 6 and could be extended to other packet specifications as they become available. The extracted Data Database contains information taken from the header as well as the payload that can be useful in anomaly detection.

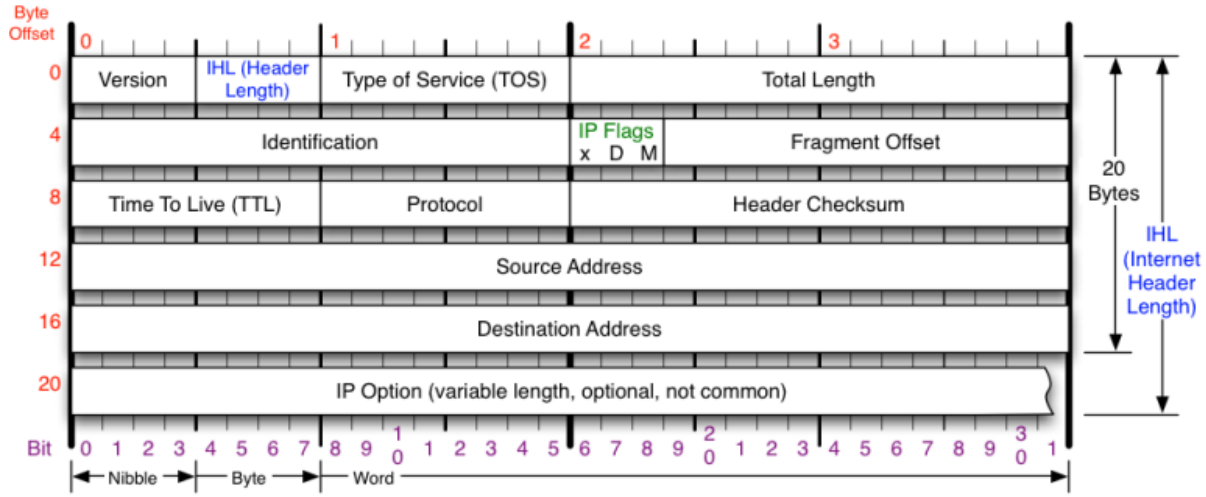
209 – Extraction and Filtering is performed on the pack data as it is received in the Packet Collection Database. This component analyzes the packet and extracts data to be passed to Layer 2 for storage. Data are extracted from the packet header and from the payload. The packet header information as shown above or for IPV6 does not contain the actual information which the packet is intended to convey. Nonetheless, the header contains information that can be useful in detecting a cyber attack. For example, there may be errors in the way the packet is formed, the source and destination addresses may be unfamiliar, or the timing may be off as when a meter sends data at too high a rate or outside the scheduled time. This component does not check for anomalies, it simply extracts the data that will be used for analysis leading to anomaly detection. Only data that will be used are passed to the Extracted Data Database, so this module performs a filtering function as well as extraction. Instruction on what information to extract is maintained in a configuration file (210).

The payload in the packets processed by Essence will consist of data in one of many messages and engineering protocols used in electric utility operations. This model will extract specified data (specified in 210) and store in the Extracted Data Database. Again, data of types not listed in the configuration file are not stored. Protocols processed by Essence may include:

- MultiSpeak Version 3
- MultiSpeak Version 5
- DNP/3
- IEC 6087—1-101
- IEEE 1379
- IEC 62351-5
- IEC 61850
- IEC 61968

- IEC 61970
- Modbus

It is presumed that this module will be extended periodically to include additional and new protocols.



Version

Version of IP Protocol. 4 and 6 are valid. This diagram represents version 4 structure only.

Header Length

Number of 32-bit words in TCP header, minimum value of 5. Multiply by 4 to get byte count.

Protocol

IP Protocol ID. Including (but not limited to):

1 ICMP	17 UDP	57 SKIP
2 IGMP	47 GRE	88 EIGRP
6 TCP	50 ESP	89 OSPF
9 IGRP	51 AH	115 L2TP

Total Length

Total length of IP datagram, or IP fragment if fragmented. Measured in Bytes.

Fragment Offset

Fragment offset from start of IP datagram. Measured in 8 byte (2 words, 64 bits) increments. If IP datagram is fragmented, fragment size (Total Length) must be a multiple of 8 bytes.

Header Checksum

Checksum of entire IP header

IP Flags

x D M

x 0x80 reserved (evil bit)
D 0x40 Do Not Fragment
M 0x20 More Fragments follow

RFC 791

Please refer to RFC 791 for the complete Internet Protocol (IP) Specification.

210 – The Configuration File (Config file in drawing) contains information on what data should be extracted from the packet for further processing. Specifications include the field name and, optionally, time intervals, source addresses, destination addresses, and source:destination pairs for which data should be extracted. If these qualifiers are omitted, then the field is extracted for all values of time, source, and destination.

211 – A Configuration File Editor is used to enter, edit, and delete specifications from the Configuration File. This editor can be implemented as standalone code or as part of a general Essence interface.

212 – Data Dictionary – a data dictionary which defines and names all fields are used the Configuration File Editor (212) and maintained by the Dictionary Editor (213). The fields in the dictionary are as follows:

- Field Name (character)
- Field Description (text)

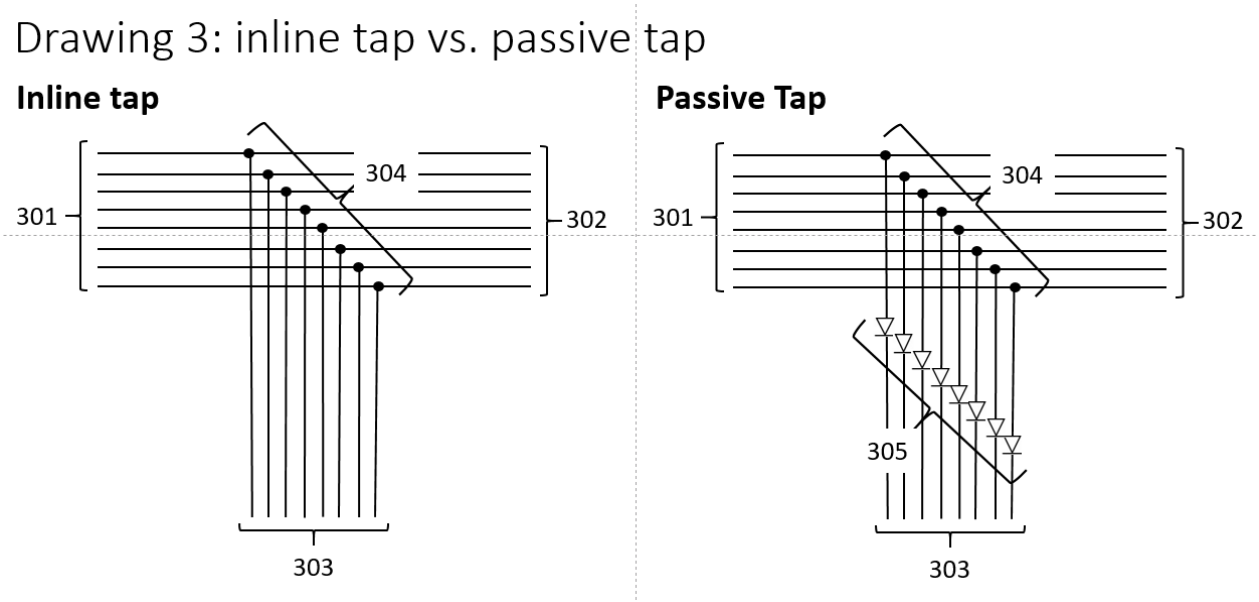
- Date Active (date)
- Date Retired (date)
- Person Entering data (character)

213 – The Dictionary Editor is used to enter, edit, and delete entries from the data dictionary and to produce reports on the dictionary contents. This serves as the reference, across all Essence layers for interpreting the data being processed. The fields are as follows:

- Data value name
- Description
- Data type
- Format is specified
- Temporal coding

214 – Element 214 is an Application Program Interface (API) which transfers data from Essence Layer 1 to Essence Layer 2.

Drawing: Inline Tap vs. Passive Tap



Drawing 3 shows the difference between an Inline Tap (Element 202 in Data Collection Drawing) and a Passive Tap (Element 203 in Data Collection Drawing). These taps both capture signals from an Ethernet cable. An Ethernet cable consists of eight wires in four pairs. There is one pair for sending information, one pair for receiving, and two bidirectional pairs. This allows for higher speed of concurrent send and receive. The inline tap simply splices into the electrical wires. The passive tap introduces a data diode. A data diode is a component that allows data to flow in one direction only, from the device to Essence.

301 – Data flow into the tap from 301 from sensors or other utility devices.

302 – Data flow out of the tap to the utility database, SCADA or other technology at the utility used for operations.

303 – A duplicate of the data passing through the tap are send to Essence via another Ethernet cable labeled 303.

304 – The connection between the through cable (301 to 302) is connected to the Essence interface cable in a way that ensures electrical continuity. Typically this is accomplished by means of intersecting traces on a circuit board but can be accomplished by means of solder or a mechanical connector.

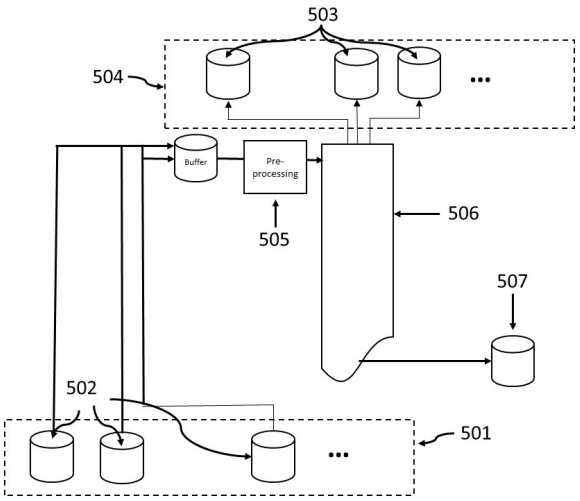
305 – This element is a data diode which allows the movement of electrical signals from the connections (304) to Essence (303). There are several means of implementing such a diode. Any reliable means is acceptable. One method is to encode the electrical signal as light which is shined on a light collecting diode which converts it back into an electrical signal. The use of components that only emit light and only capture light means that no signal can enter the trap from Essence (303).

In this example, a separate diode is used for each of the individual wires. A more common implementation is to use a commercial data diode which implements these connections internally, so from a user perspective the array of eight diodes can be installed in line through RJ45 (Ethernet connectors) on the outside of the box containing the array of diodes.

Multiple instances of the data collection layer (see Data Collection Drawing) can be implemented. These can be implemented independently on a single utility. An example of this would be implementation of separate systems on different feeders of a utility where the feeders operate with a measure of independent control over unconnected data systems. Another example would be two utilities choosing to share a single instance of Essence Layer 2, 3, 4, and 5. The interface between Layers 1 and 2 is implemented as an API.

Layer 2: Information

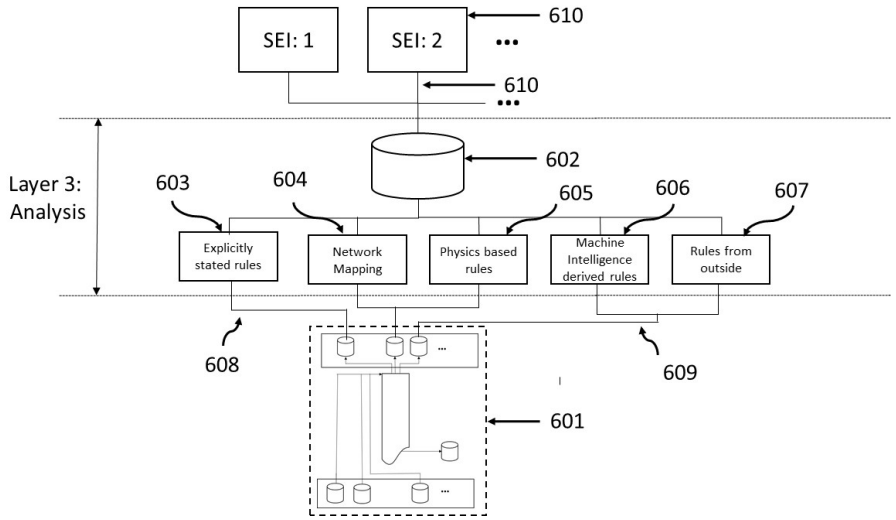
The information layer was implemented using Cassandra, a high-performance open-source database. The architecture was highly innovative, designed to provide openness and high performance.



In the drawing above, data arrive from Layer 1 through a series of small databases (502) associated with multiple collection points. Each of these pushes a single record to a central database (506) through a buffer and a filter (506) that will eliminate records that are of no analytical interest. The central database should be thought of as a paper tape and is implemented in memory. New values are inserted at the top and old values are pushed down and ultimately migrated to magnetic storage (507). Different applications and groups of applications in Layer 3 have different data requirements. Individual applications do not need all of the data being processed. Small databases (503) are established for each application or group of applications which share common data requirements. Data from the central database are distributed to the application-specific databases by means of a publish/subscribe method.

Layer 3: Application

The application layer is designed so that different organizations and individuals can develop their specific applications and plug them into the Data Layer through the application-specific databases. In the Essence project, we developed a rules manager, shown previously, which composes rules in JSON. These rules can come from multiple sources as shown in the drawing below.



Data from Layer 2 (601 in drawing above) goes through the application-specific database. Then, applications of different types apply anomaly detection rules. These include:

- Explicitly defined rules, specified by an expert
- Network mapping to identify normal patterns of communications, including origin and destination pairs, the protocols used, and typical timing
- Physics based rules – modeling power flow to identify inconsistent or unrealistic telemetry
- Machine derived rules – rules derived by modeling observed behavior to derive a classification engine
- Rules by others – Other organizations can send JSON rules using a common data dictionary.

Layer 4: Decision

The decision Layer was implemented through an integration into Alien Vault, a commercial Security Event and Incident Management (SIEM) tool. NRECA's analysis and discussions with commercialization partners convinced us that broad commercial adoption would be more successful if done as part of a general-purpose security management tool rather than as another product. This work was done in conjunction with SEDC (South Eastern Data Cooperative), a major supplier of IT software and services to several hundred electric cooperatives. The advantages of Alien Vault at SEDC are that this approach:

- Provides a single interface to present information from Essence and SEDC MSS,
- Conducts cross-correlation of information collected by Essence and SEDC MSS,
- Decreases time required to deploy Essence appliance (if SEDC MSS or AlienVault is already in service at an electric utility)

This work with Alien Vault, considered proof-of-concept, is discussed below.

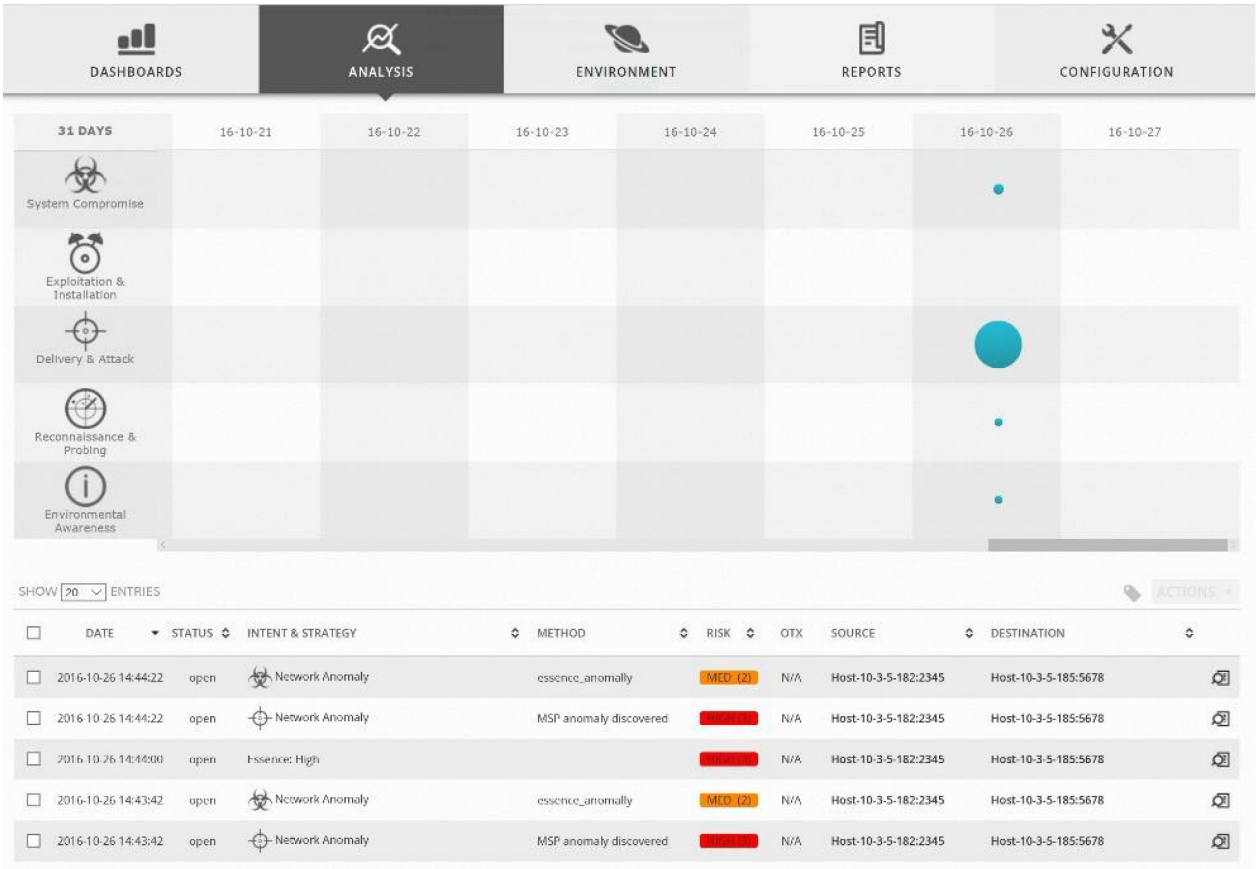
The work in this layer was structured into four steps

1. Data source plugin. The plugin was modified by SEDC to enable access to the MySQL database on Essence (The plugin will provide the information to be used in the next steps of analyzing the anomaly (policies, correlation).
2. Policies. In the AlienVault architecture, each event is subject to a specific policy which determines further actions - such as conducting risk assessment or storing event information in the SQL database. A new custom policy was built by SEDC in the AlienVault system to conduct the following actions on events received from Essence: -
    a. Risk assessment
    b. Logical correlation
    c. Cross correlations
    d. Storing event in SQL database

    This new policy was tested using the four levels of severity for Essence events (Informational, Low, Medium, High).
3. Logical correlation. Each event can be analyzed by logical correlation and the appropriate event type can be assigned (for example System Compromise) – this process helps operators with system status assessment. Logical correlation is included in this proof of concept as shown in the table below:
4. Cross-Correlations. SEDC MSS has an advanced functionality called cross-correlations, which allows the program to correlate information provided by two different sources. Basic cross correlation example shall be included in this proof of concept, for example correlating Essence alarm with MSS alarm showing unsuccessful login into computer involved in MultiSpeak communication.

Essence Severity	**AlienVault Intent**
Informational	None
Low	Environmental Awareness
Medium	Reconnaissance and Probing
High	Delivery and Attack

The cross-correlation is shown in the following graphic from AlienVault's console:



In this task, the Layer 4 work:

- Demonstrated that all requirements (goals) can be achieved.
- The Essence plugin and AlienVault will require some adjustments for specific networks.
- This type of integration provides additional value to both Essence and SEDC MSS.

Layer 5: Action

The Action Layer was implemented only as a prototype. A capability was developed to take action using software-defined networking (SDN) to isolate a communication from a suspect source. The user was offered the option to continue using data from the source, to sever communications from the source, or to embargo data from the source for forensic analysis or use after further analysis. This was implemented using the OpenDaylight SDN specification. Unfortunately, we did not find the then-current implementation of OpenDaylight to be sufficiently robust and complete to proceed with commercial development.

The work in the project demonstrated that, at a future date when SDN is more mature and widely deployed, networks can be reconfigured in response to cyber-attacks. In the future we intend to engage with Schweitzer Electric Laboratories (SEL) who makes a reliable SDN switch and controller.

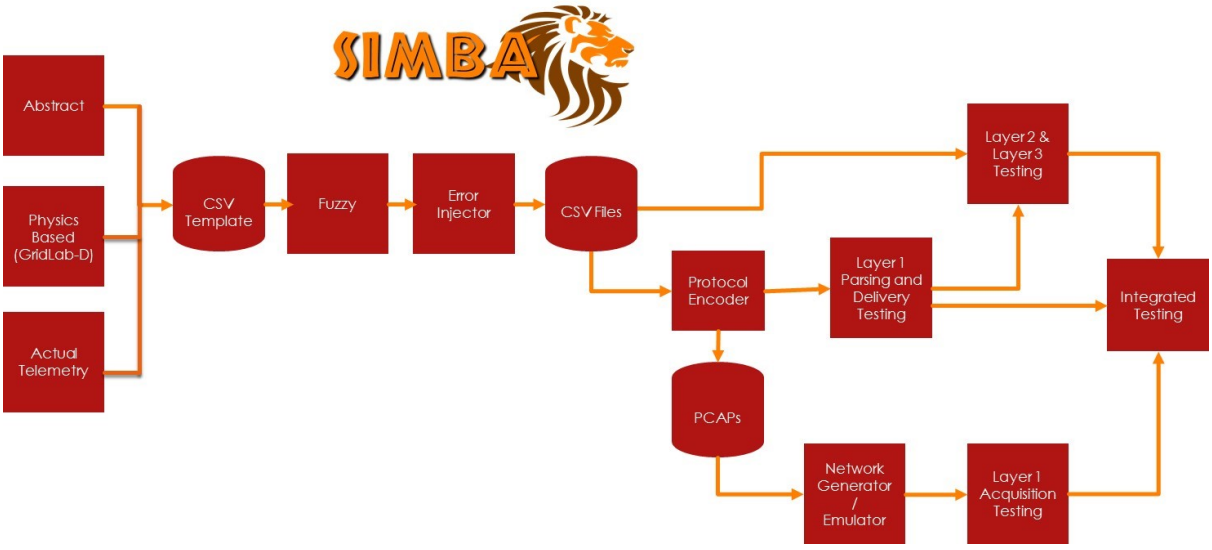
Accomplishments of Task 4 – Network System Design

The original network design used locally networked computers. Since then, the entire system has been migrated to the Amazon Web Services cloud.

Accomplishments of Task 5 – Laboratory Testing

Laboratory testing of Essence presented a considerable challenge. Each of the system components was tested in the conventional ways. Testing the entire system, through its ability to learn, is much more challenging. A testing system was developed for this purpose. Called "Simba", the testing harness has the capability to generate realistic network traffic at extremely high speeds and stream these into Essence as network traffic at much faster than real time (MFRT). The system can introduce errors in packet formation, payload formation and values, and in timing. It is intended to process a year's data 10,000x real time, so that a year's operation can be modeled in less than one hour.

The structure is shown below.



The starting point of the Simba system, prior to error injection is either:

1. A file of value generated from the abstract; e.g. value in the range 114-125 for meter voltage readings,
2. Telemetry captured in actual utility operations, and
3. Values modeled with a power-flow model.

The value from these sources will comprise the payload value for the synthetic telemetry.

Statistical variation is then added (Fuzzy) and payload errors are injected (Error Injector). The output of these are comma delimited files. A library of these is stored. These data can be use directly for database

(Layer 2) testing and to test rules. The data can also then be encoded as packets and sequenced to create packet capture (PCAP) files. A library of PCAP is maintained. Files are drawn from that library and errors can be injected. A model of the network is constructed and then the PCAPs are played from modeled nodes emulating network traffic. This allows for testing of the capture technology.

Accomplishments of Task 6 – Field testing

Field testing was done at five electric utilities, but Simba testing ultimately proved more useful, as the data rates in the field are low enough that real-time testing does not stress the system, and there are few to no anomalies to detect during a reasonable period.

Accomplishments of Task 7 – Commercialization

NRECA has, thus far, engaged with four companies to build commercial managed services based on Essence. The basic structure is that there is "Essence Core" technology which consists of Layer 1 (Data) and Layer 2 (Information) and the Layer 3 code which applies the rules and generates alerts. This code will be share by all Essence commercialization projects. It may be run on the same cloud infrastructure.

Each company will build their own Layer 3 rules set, Layer 4 Decision interface and, optionally the Layer 5 Action capability. The agreed partners are:

- South Eastern Data Cooperative,
- N-Dimension,
- Milsoft, and
- National Rural Telecommunications Cooperative.

A meeting is scheduled with the National Information Solution Cooperating (the largest software and IT service provider to co-ops) on May 25th and 26th to discuss a partnership for commercialization. The barrier is that NISC competes with the current partners. This will be a complex negotiation.

5 PRODUCTS

5a Publications, conference papers, and presentations

SEDC announced the integrated product at their annual meeting in August. Announcements by the other partners were made at NRECA's largest annual meeting – Tech Advantage, in February of this year. Technical details and a schedule for launch will be made by N-Dimension, Milsoft, and NRTC at Milsoft's annual meeting in June of 2017.

5b Websites or other Internet sites

There are more than 50,000 references to NRECA's Essence on the Internet. The commercialization partners have already or will, shortly, announce Essence on their web site.

5c Collaborations

Essence work is being continued with all of the commercial partners contributing labor and infrastructure, and DARPA is conducting substantial R&D with NRECA to extend the technology. An updated version, built with DARPA funding will be released on 4 July. It improves on the base performance. Code from this version will comprise Essence Core for commercialization efforts.

5d Technologies

- MultiSpeaker™ software,
- Passive network hardware capture technology,
- Virtual tap for capturing traffic in VMWare installation,s
- Network graph annotation technology,
- Integration with Alien Vault SIEM,
- Software defined network controls using OpenDaylight,
- Distributed, heterogeneous, dynamically reconfiguring database
- Large network generator (Simba)

5e Patents, IP, Licenses

Four patents are in process –

- Essence for cyber security applications,
- Essence for broader industrial control system applications,
- Simba for cyber security testing, and
- Simba for broader industrial control systems applications.

Attorneys have been engaged to carry this forward. They are considering now whether the four patents, as drafted, should be combined into one for Simba and one for Essence.

6. Computer Modeling and Software

Software for the first version of all layers has been posted on GitHub and is offered with a BSD 2-Clause license.