Failure Modes and Effects Analyses (FMEAs) are widely used in safety and mission critical applications. The U.S. Department originated the technique in the 1960's and institutionalized them in multiple reliability and safety standards. These include

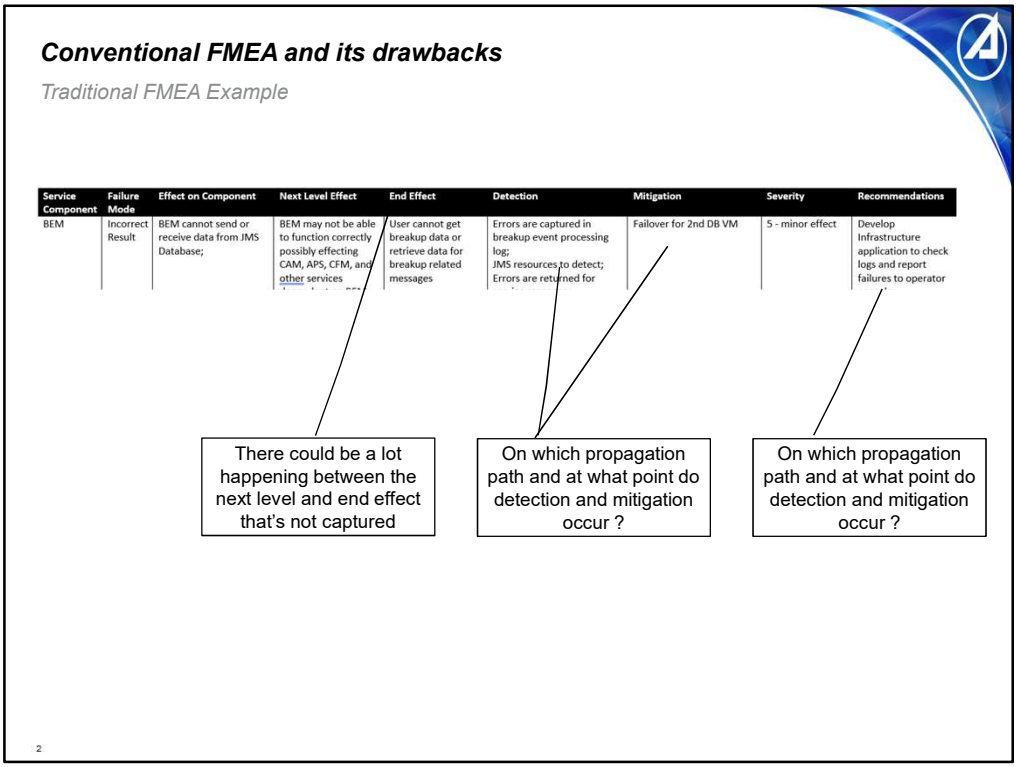
- Defense (MIL STD 1629A, GEIA STD 009, and MIL STD 882E)
- Avionics (SAE ARP 4754, ARP 4761, and AARP 5580)
- Automotive (SAE J1739)
- Medical Devices (ISO 14971 risk management, ISO 60812 FMEA, FDA Guidance for Industry, Q9 Quality Risk Management)
- Nuclear Power Reactors, Space Systems, Industrial Process Control,

The systematic and thorough analysis approach mandated by FMEAs have resulted in their application for cybersecurity in information systems (Refs. 1, 2, 3), often using variations such as F(I)MEA (where I is intrusion; Butler 2006) for FMVEA (where V is vulnerability ; Schmittner 2014). For network management and system management, FMEAs provide a method of correlating end effects with causes and as such, can be an important aid to intrusion detection as well as for incident response.

Ideally, FMEAs should be done at multiple stages in the development process to identify failure detection and recovery deficiencies as early as possible and to take corrective action when it is still feasible to do so from the perspective of cost, schedule, and technical solutions. The practice of developing FMEAs "early and often" in the design process has not been feasible using traditional manual techniques in most development programs because of their cost and skilled labor requirements. Hence, there has been significant interest in the development of techniques based on SysML and AADL (see for example, David, 2010; Larson, 2013) and an Object Management Group is now developing a standardized profile to be used for the development of FMEAs.

This paper combines (1) application of FMEAs for integrated cybersecurity, reliability, and safety analysis and (2) use of SysML for the automatic generation of such analyses and demonstrates the application of this approach to a Supervisory Control and Data Acquisition (SCADA) information network.

978-3-903176-15-7 © 2019 IFIP



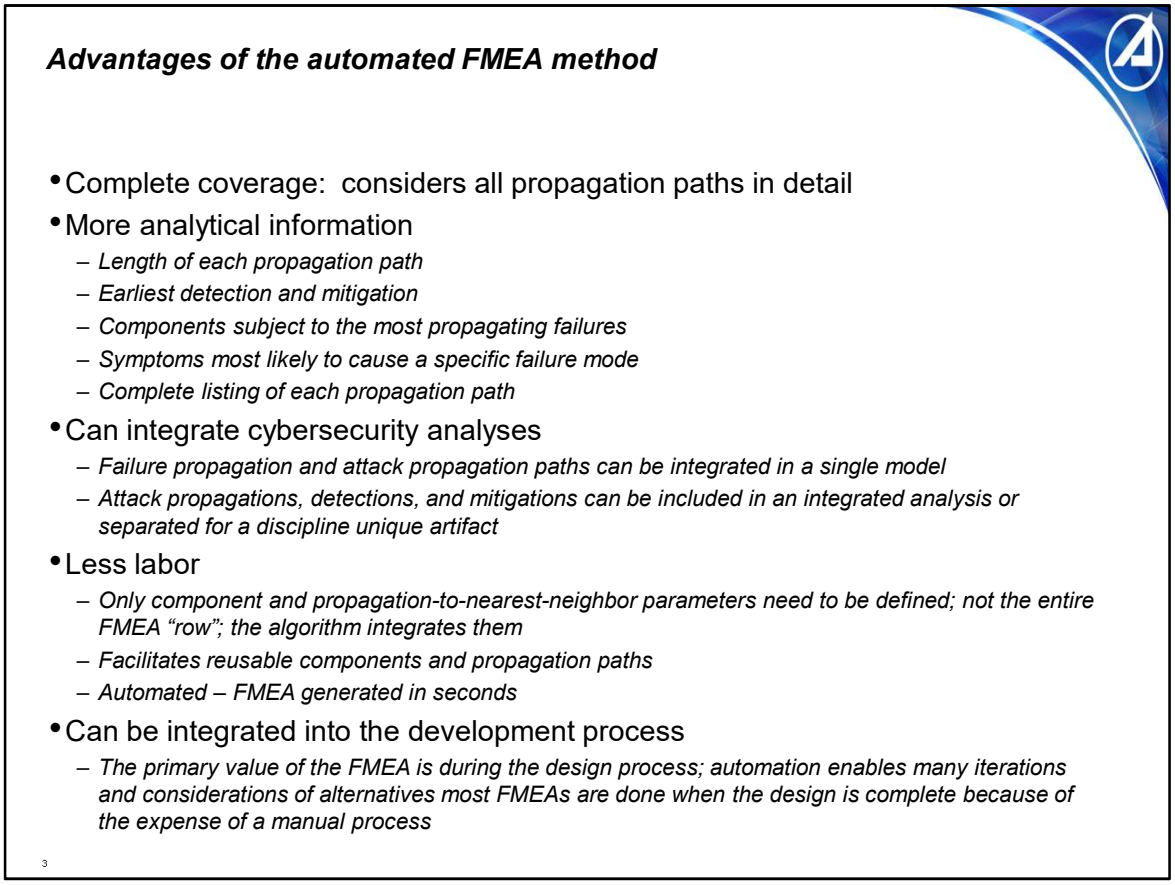
This chart shows a row of a typical FMEA. The following is the underlying conceptual framework (ontology)

- **Components** have a number of *failure modes*
- **Failure modes** *propagate* from *component* to *component*
- **Failure propagations** are the occurrence of one *failure mode* causing another
- **Failure transformations** describe *failure modes* changing type as they *propagate*
- **System level effects** are *transformations* that affect the function or operation of the system as a whole
- **Severities** measure the harmful consequences of *system level effects*
- **Detections** are methods for discovering a failure once it has occurred
- **Mitigations** are methods of reducing or negating the effects of a failure occurrence

The column of the left is the component under analysis which is paired with the failure mode. This is the fundamental unit of analysis in a typical FMEA. The table then shows the effect on the component, the next level effect, and the end effect. The next two columns describe the handling of the failure and effects through detection and mitigation. The second to rightmost column is the severity rating (in this case, 1 through 5 with being most severe) and comment which might be a recommendation for a system improvement, an notation of uncertainty in the analysis requiring further resolution, or an assumption. The comment column is often the most important product of the FMEA because it may lead to the addition of new mitigations that result in a safer and more secure system (if the analysis is done at a time when design changes are still feasible). There are variations on this general format including a separate column for causes and additional columns for a probability assessment which, when combined with the severity provides a measure of risk, or Criticality, in which case, the FMEA is sometimes referred to as an FMECA

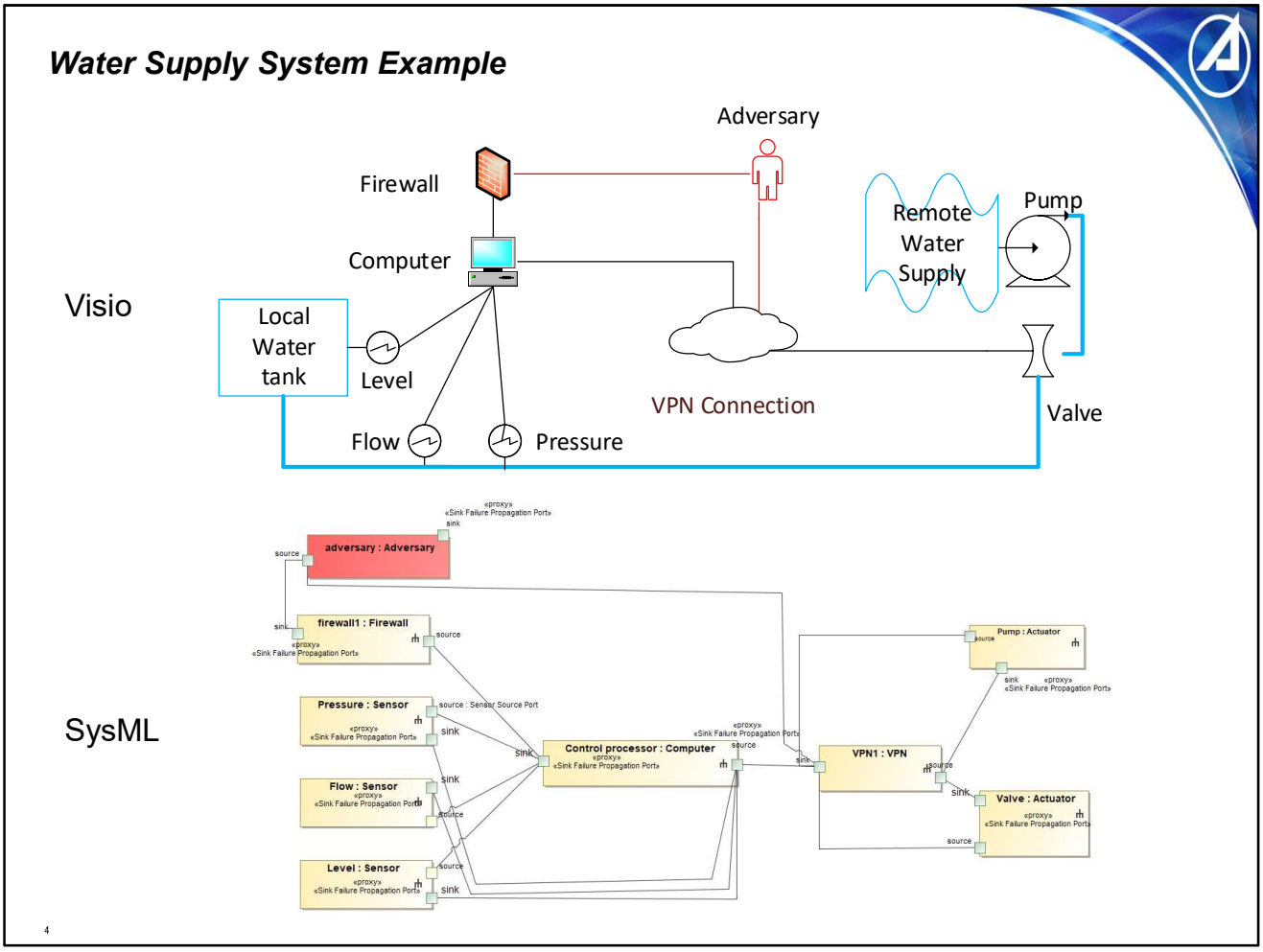
There are problems with this approach including:

1. The commitment expertise necessary to effectively formulate failure modes and understand the system responses to them, and the amount of time (mentioned earlier)
2. The fact that only three levels of effects (immediate, next higher, and end) are considered; there might be 5 or 10 additional effects which are not explicitly considered in this analysis
3. The fact that because not all failure propagation paths are considered, the detections and mitigations may effective on some but not all propagation paths. This is particularly true for cyberattacks. As a result, an FMEA that considers only one failure propagation path for each component and failure mode will not suffice as a tool for cybersecurity analysis.



As we will show later, our approach overcomes these limitations by enumerating and analyzing all propagation paths for each failure mode/component/cause triple. For each propagation path, our software calculates total length and the earlier point of detection and mitigation (i.e., closest to the postulated failure origin). We also tabulate the number components subjected to the most failure modes, symptoms linked to each failure modes, and list all propagation paths. This listing and enumeration of all propagation paths enables the complete identification of cyberattack paths, and where defenses (either in terms of detections and mitigations or preventative/protective measures) can be evaluated.

A major advantage of our automated approach is that each analysis can be performed in seconds even for larger models, thereby enabling multiple analyses to be performed in the course of each design stage and making the FMEA evaluation integrated into the design activity.

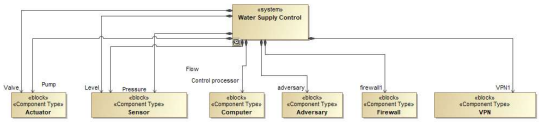


This chart shows a notional water supply system which representative of a Supervisory Control and Data Acquisition (SCADA) systems that we will use as an application example. The upper chart shows an informal representation in Visio. The security boundary is defined at the top by the firewall. On the left is the local control center which consists of a computer system with three sensors: level, flow, and pressure. At the right are the remote parts of the system consisting of a valve and a pump located at the water supply (a lake or a river). The data connection between the computer and the actuators (pump and valve) is through a Virtual Private Network. The water supply and tank are connected by a pipeline shown as a blue line. The level measures that amount of water in the local tank and when it falls below a set point, the computer system initiates commands to start up the pump and open the value. Flow and pressure measurements on the pipeline are used to modulate the speed of the pump. We assume that an adversary can attack the system either through the external face of the firewall or through the VPN.

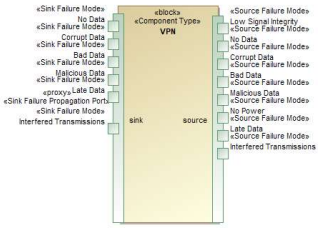
The lower chart shows a SysML Internal Block diagram (IBD) that represents the same system. The parts are defined by names and data types. The data types enable reuse of the failure modes, transformations, and propagations that are defined for each type of component. The ports on the blocks (represented by small squares) and lines are not normal ports and connections for information or item flow; they are specialized ports and paths for failure propagations. Thus, for example, there is no data or flow connection normally between an adversary and the firewall or between the adversary and the VPN. However, there is an attack path and hence, the connection. As we will show later, both the connections and the ports are highly specialized for failure propagation and transformation. However, at this level of detail, those structural features are hidden.

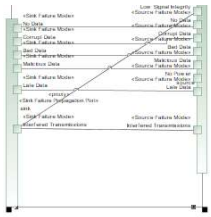
Automated FMEA Generation Procedure

- Define failure propagations and transformations in SysML
- System described using standard SysML constructs
- Once system is modeled, output is automatically produced

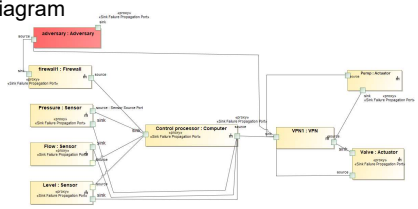


1. Defining the System with a Block Definition Diagram

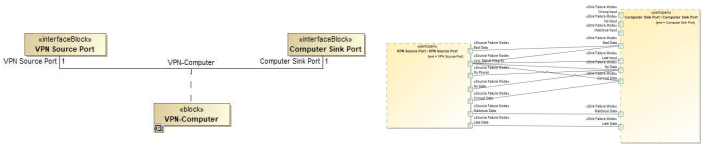




2. Defining the failure propagations and transformations within a component



3. Defining the propagation paths with a System Internal Block Diagram



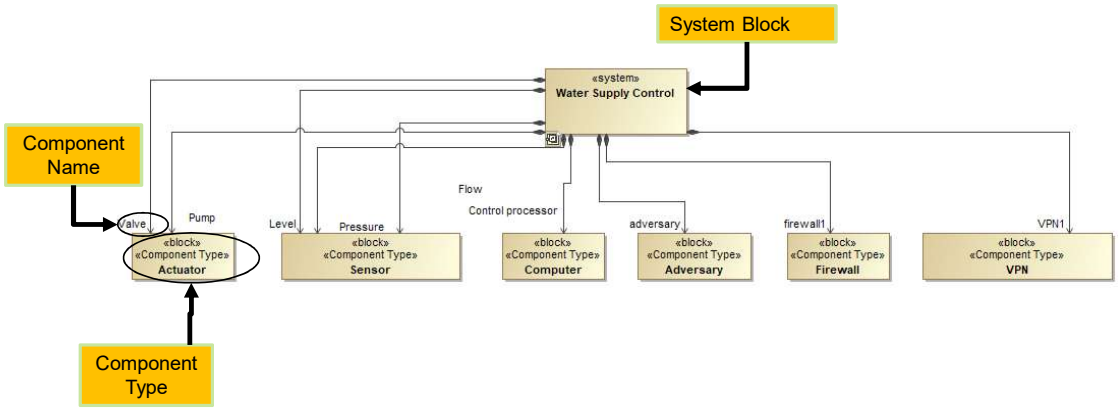
4. Defining Inter-component propagations and transformations

5

This chart summarizes the automated FMEA generation process that will be described in more detail in the following charts. It consists of 6 major steps

1. Defining the system components to be included in the analysis using a SysML Block Definition diagram
2. Creating a SysML Internal Block diagram that identifies the propagation paths among the components (the diagram shown earlier).
3. Defining the inter-component failure propagations along each of the paths using SysML association blocks and failure transformations using internal block diagrams
4. Defining the internal component propagations using block definition diagrams and transformations using internal block definition diagrams

1.Defining the system components to be included in the analysis using a SysML Block Definition diagram



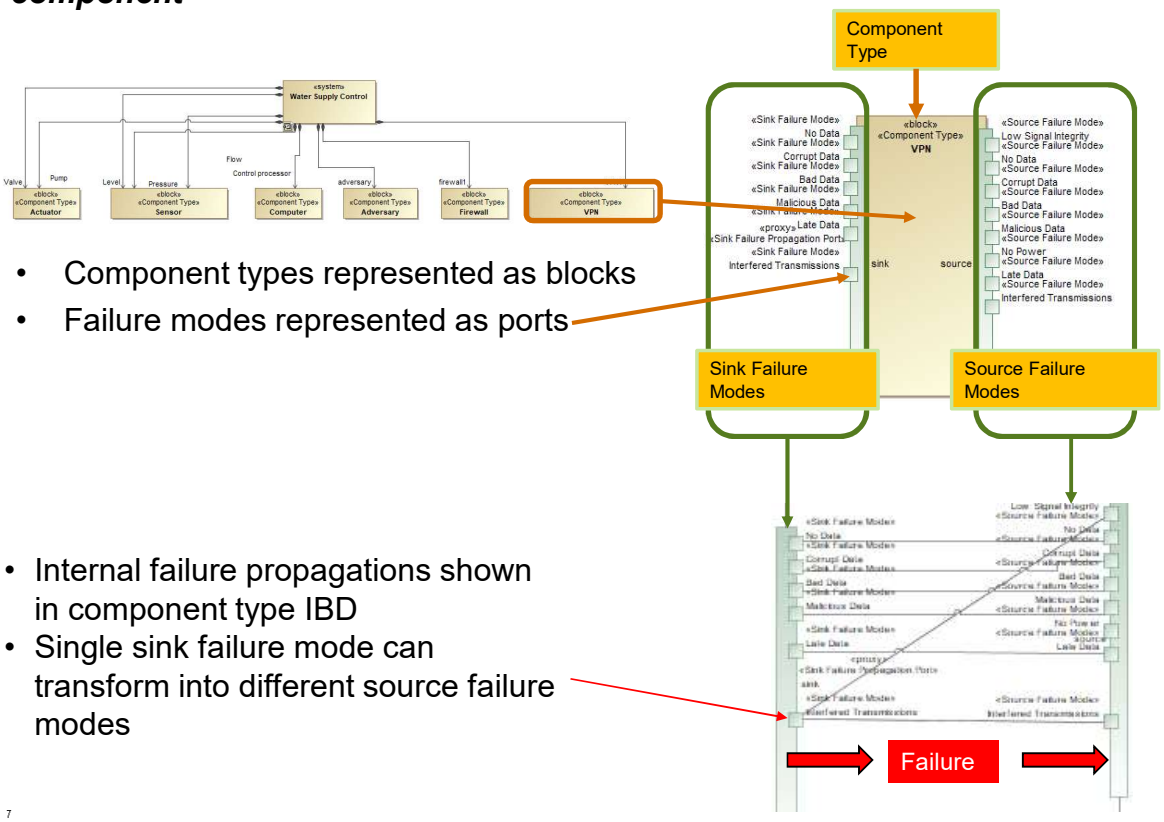
- System represented by top-level block
- Component types connected to subsystem through the directed composition relationship
- Components are instantiated from component types

6

In the first step in the procedure, we define the system using a SysML Block Definition Diagram. If the system under analysis is being developed using an Model Based Systems Engineering (MBSE) approach with SysML, this diagram will have been produced already. The major concepts depicted on this diagram are

(1) The water supply system is composed of 5 major system component types: actuators, sensors, computers, firewalls, and a VPN. For the purposes of analysis, we have added a 6th "component" type: an adversary that will be used for the cybersecurity analysis
(2) These component definitions will be instantiated into components using the names (roles) shown at the ends of the arrows that join the top level block (water supply system) to the lower level blocks (e.g., the actuator at the level of the diagram)

2. Defining the failure propagations and transformations within a component



- Component types represented as blocks
- Failure modes represented as ports

- Internal failure propagations shown in component type IBD
- Single sink failure mode can transform into different source failure modes

7

The next step is to define the failure propagations within each of the component block definitions that we defined in step 1. This chart shows the overall relationship between the BDD are the more detailed failure propagations and transformations for the purposes of the FMEA that we will describe in the next few charts.

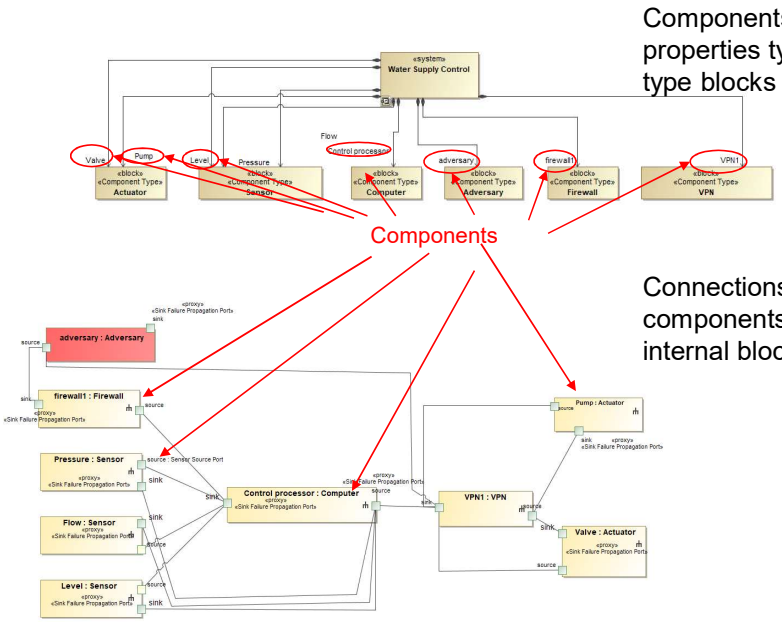
Two kinds of SysML ports are define: (1) generic failure propagation ports called sink ports and source ports and (2) specific source and sink failure mode ports that are nested within their respective propagation ports. It is not necessary to create these ports manually. The model and tool we have created can perform this action using predefined stereotypes.

When creating designs using SysML, the blocks identified in the Block Definition Diagram (BDD) are defined with many different properties (e.g., weight, power, capacity, etc.). In order to perform the FMEA, we must define their failure modes. We do this by defining two failure propagation ports: an input or "sink" port, and an output or "source" port. As we will we show in the next chart, failures propagate paths defined by connections from sources to sinks.

Also, another important point is that these failure propagations are included in the block definitions. That means that they can be reused in future analyses thereby not only saving labor but also enabling standardization.

Once the input (sink) and output (source) ports have been defined, the failure propagations are defined using a SysML internal block diagram (upper diagram) that represents both the component and the outside ports of the block definition (lower diagrams). Horizontal lines connecting the input and output ports represent propagation. Diagonal lines represent transformations. In this diagram, interfered transmission on the VPN can be transformed into the absence of data.

3. Defining the propagation paths with a System Internal Block Diagram

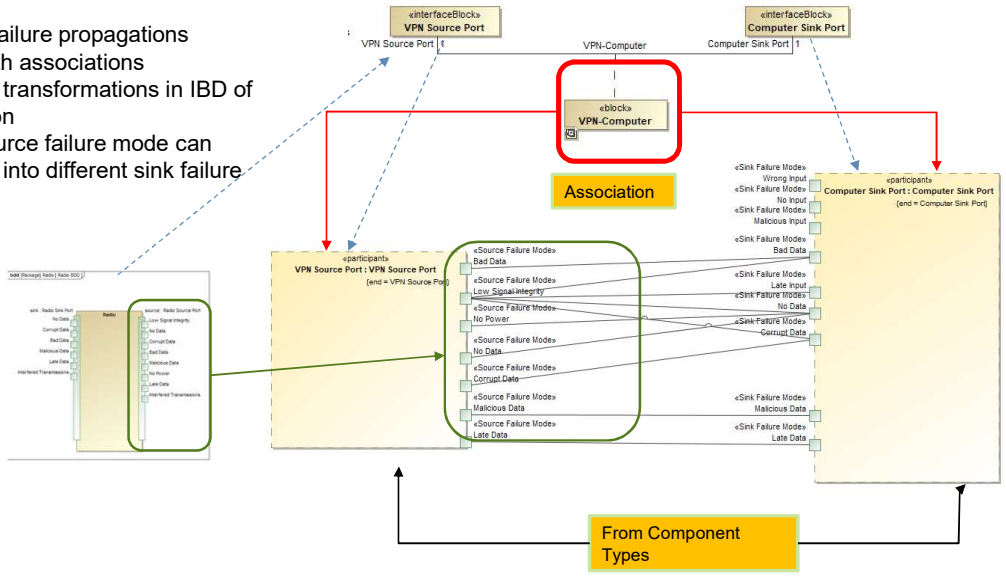


8

For the next step, we describe the failure propagation paths among the components using a SysML Internal Block Diagram (IBD). This diagram shows the relationship between the component definitions in the BDD and their instantiations in the IBD both of which were shown earlier. Notice that on the IBD, the IBD shows the failure propagation ports.

4. Defining Inter-component propagations and transformations

- External failure propagations shown with associations
- Individual transformations in IBD of association
- Single source failure mode can transform into different sink failure modes



The final step is defining the failure propagations and transformations between components. This chart describes the SysML association block, the modeling construct used to represent the details of the failure propagation. The top diagram shows the definition of the connection which shows the source port (VPN in this case) connected to the sink port (the computer in this case). The name of the connection (or "association" using SysML terminology) is VPN-Computer. The lower diagram shows the propagations and transformations between the components represented in a SysML internal block diagram. As was the case with the intra-component failure propagations and transformations, failure propagations are represented as horizontal lines connecting source and sink ports; failure transformations are represented as diagonal lines.

This chart also shows how the association block relates to the internal block diagram below, and how the propagation source ports on the left-hand component related to the propagation sink ports for the right hand component.

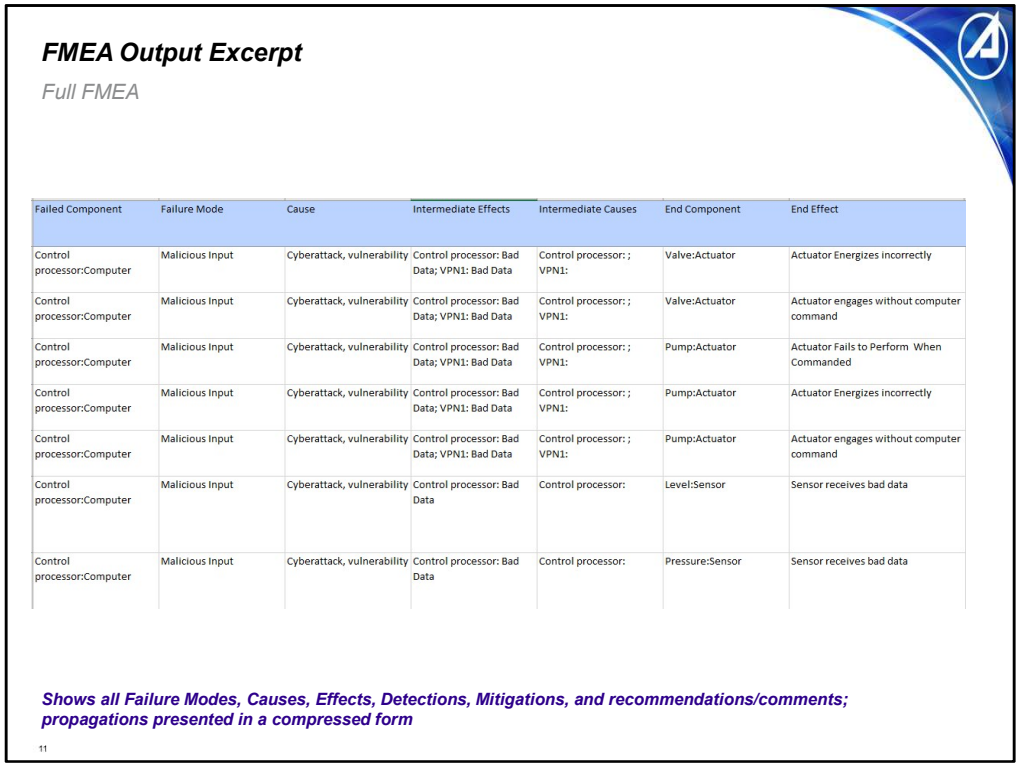
FMEA Output

Table	Description and Use	Water Supply System Results
Full FMEA	List all FMEA information in SysML model Rows represent individual failure propagation paths	There are 1110 propagation paths with unique originating components, failure modes, causes, propagation steps, and end effects (with a conventional manually generated FMEA, there would be only 37 rows)
Failure Modes and Effects Summary	Provides both qualitative and quantitative data about each failure mode and effect Useful for prioritizing failure and cybersecurity resources by identifying system components with the highest number of failure modes, undetectable or unmitigated failure modes, and long propagation paths	The VPN is the component with the most failure modes, actuator failure modes have the highest proportion of severity 1 events, CRCs and redundancy checks are the most often used detection mechanism, Retry is the most common recovery mechanism. Malicious Data is the failure mode that is most often not detected and has the greatest severity effects
System Effects Summary	Provides analysis of all system effects in system Useful for determining undetected, unmitigated, or unprotected system effects	The VPN is the component with the largest number of severity 1 failure modes Actuators (pump and value) and the control processor are also significant contributors to Severity 1 failure modes
Diagnostics	Matrix of system effects versus their causes Capable of determining probably causes of system effects	The VPN is the single component most likely to be the cause of malfunctions in the actuators The control processor can be a cause of all system level effects identified thus far
Propagation Description	Rows represent individual failure propagation paths Each cell in a row lists detailed information about a single failure propagation hop	There are multiple propagation paths for which there is no protection against a cyberattack; measures for failure detection and mitigation should be evaluated to determine if there is any effect

10

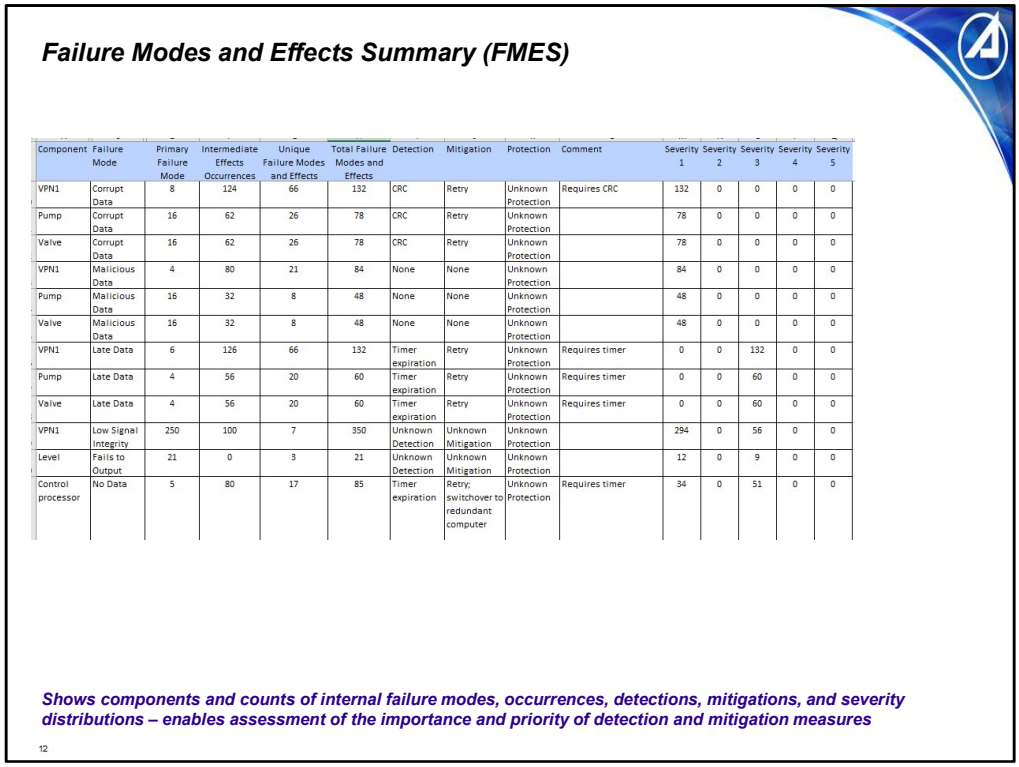
The SysML FMEA tool outputs five tables that will be described in the following charts. The tables are output as tabs in a Microsoft Excel spreadsheet to enable subsequent formatting, summarization, and visualization.

This chart describes the tables and summarizes some of the significant results emerging from the application of the automated FMEA to this system. It demonstrates the depth of insight and actionable information that can be gathered – much more than from a conventional FMEA. Furthermore, this information can be acquired early enough in the design that it is possible to make changes that can improve its robustness and resiliency.



This chart shows a portion of the full FMEA that is generated by the tool. For the example water control system described here, there were a total of 1110 rows to account for all combinations of failed components, failure modes, causes, and propagation paths. Also, this chart shows only 7 of the 22 columns that are produced by our plug-in. The complete set of columns are

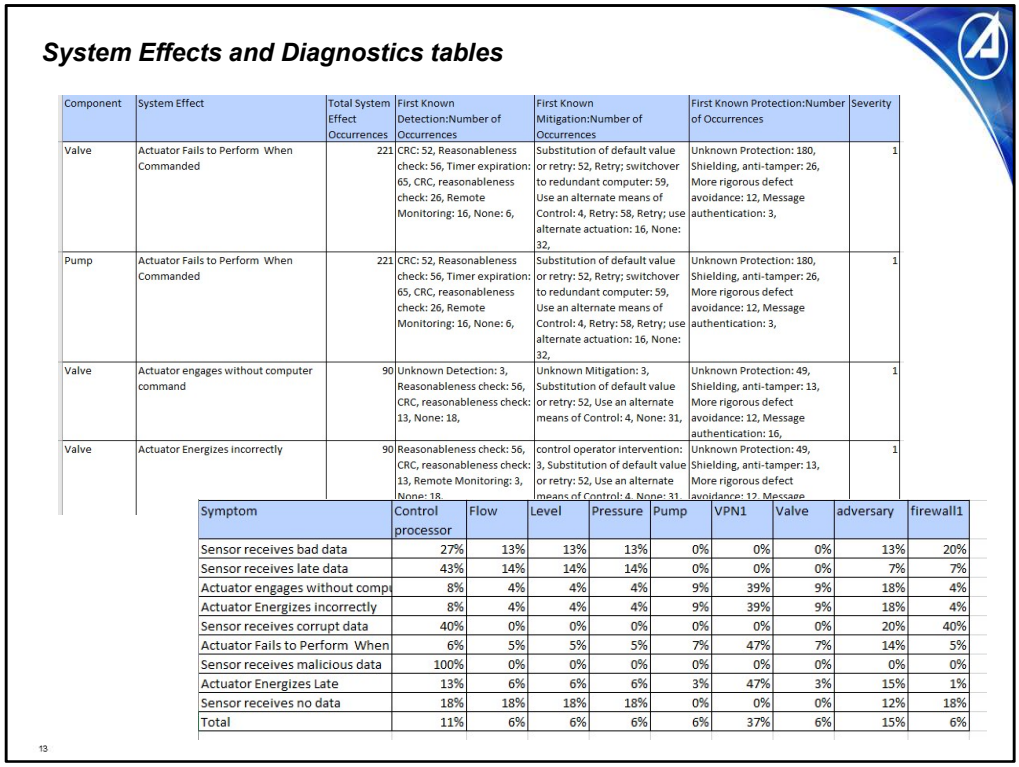
- **Failed Component:** Identification of the specific component and component type
- **Failure Mode:** Identification of the failure mode
- **Cause:** Cause of the failure mode. If there are multiple causes, each cause is listed on a separate line because the protection/prevention measures would differ
- **Intermediate Effects:** Identification of each of the effects (secondary failures) as the primary failure propagates through the system until its end effect. Note that this propagation path is further detailed in the Propagation Description table
- **Intermediate Causes:** The causes associated with each of these intermediate effects
- **End Component:** The component at which the failure propagation terminates (end effect)
- **End Cause:** The cause of the failure at the end component
- **Severity:** The severity of the end effect
- **Severity Comment:** Explanation or uncertainty in determining the severity
- **Detection**: Detection of the end effect
- **Mitigation:** Mitigation of the end effect
- **Protection:** A protective or preventative measure to prevent the failure or cyberattack effect from occurring
- **Comment:** Explanation of the protective measure or documentation of uncertainty
- **# Propagations:** Number of components involved in the propagation from the primary failure mode to the end effect
- **First Known Detection:** First component in the propagation path at which the failure can be detected
- **# Propagations to Detection:** Number of components affected by the failure until it is detected
- **First Known Mitigation**: First component at which a mitigation of the failure can occur
- **# Propagations to Mitigation:** Number of propagations to the mitigation
- **First Known Protection:** First protective measure along the propagation path that can prevent failure propagation from occurring (particularly relevant to cybersecurity)
- **# Propagations to Protection:** Number of components involved in the propagation until the protective measure is reached
- **Intermediate Detections:** List of all detection mechanisms other than the primary and end effect detections along the propagation path
- **Intermediate Mitigations:** List of all mitigations other than the primary and end effect mitigations along the propagation path
- **Intermediate Protections:** List of all protection measures other than the primary and end effect protection measures along the propagation path
- **Intermediate Comments:** Explanation or uncertainties on the intermediate detections, mitigations, or protections



The failure modes and effects summary (FMES) is one of the two most useful outputs from our plug-in because it enables a rapid identification of the failure modes that lead to the most severe effects, components with the most failure modes, the most used detection and mitigation effects, and the distribution of failure modes by severity. The specific columns in the FMES are

- **Component:** Name or identification of the component
- **Component Description:** Description of the component (taken from the SysML block documentation field)
- **Failure Mode:** Failure mode of the component (taken from the sink failure mode)
- **Failure Mode Description:** More detailed description of the failure mode (hidden in this particular output)
- **Primary Failure Mode Occurrences:** Number of times this failure mode appears as a primary (i.e., left most) failure mode
- **Intermediate Effects Occurrences:** Number of times this failure mode appears as an effect (i.e., after the first failure mode)
- **Unique Failure Modes and Effects Occurrences:** Number of unique propagation paths in which this failure mode appears
- **Total Failure Modes and Effects Occurrences:** Sum of primary and intermediate failure mode occurrences
- **Detection:** Means of detecting this failure mode
- **Mitigation:** Means of recovering from or otherwise limiting the severity of the failure mode
- **Protection:** Means of preventing this failure mode (particularly important if it cannot be detected or recovered from)
- **Comment:** Comment containing assumptions, unknowns, or requirements associated with this failure mode
- **Severity 1:** Count of propagations in which this failure mode is associated with a highest severity (severity 1) effect
- **Severity 2:** Count of propagations in which this failure mode is associated with a severity 2 effect
- **Severity 3:** Count of propagations in which this failure mode is associated with a severity 3 effect
- **Severity 4:** Count of propagations in which this failure mode is associated with a severity 4 effect
- **Severity 5:** Count of propagations in which this failure mode is associated with a highest severity (severity 1) effect

From the FMES, we learn that the VPN is the component with the most failure modes, that actuator failures have the greatest proportion of severe failure modes, that CRCs and redundancy checks are the most often used detection mechanism, and that retry is the most common recovery mechanism. The failure mode that is most often not detected and has the greatest



Component	System Effect	Total System Effect Occurrences	First Known Detection:Number of Occurrences	First Known Mitigation:Number of Occurrences	First Known Protection:Number of Occurrences	Severity
Valve	Actuator Fails to Perform When Commanded	221	CRC: 52, Reasonableness check: 56, Timer expiration: 65, CRC, reasonableness check: 26, Remote Monitoring: 16, None: 6,	Substitution of default value or retry: 52, Retry; switchover to redundant computer: 59, Use an alternate means of Control: 4, Retry: 58, Retry; use alternate actuation: 16, None: 32,	Unknown Protection: 180, Shielding, anti-tamper: 26, More rigorous defect avoidance: 12, Message authentication: 3,	1
Pump	Actuator Fails to Perform When Commanded	221	CRC: 52, Reasonableness check: 56, Timer expiration: 65, CRC, reasonableness check: 26, Remote Monitoring: 16, None: 6,	Substitution of default value or retry: 52, Retry; switchover to redundant computer: 59, Use an alternate means of Control: 4, Retry: 58, Retry; use alternate actuation: 16, None: 32,	Unknown Protection: 180, Shielding, anti-tamper: 26, More rigorous defect avoidance: 12, Message authentication: 3,	1
Valve	Actuator engages without computer command	90	Unknown Detection: 3, Reasonableness check: 56, CRC, reasonableness check: 13, None: 18,	Unknown Mitigation: 3, Substitution of default value or retry: 52, Use an alternate means of Control: 4, None: 31,	Unknown Protection: 49, Shielding, anti-tamper: 13, More rigorous defect avoidance: 12, Message authentication: 16,	1
Valve	Actuator Energizes incorrectly	90	Reasonableness check: 56, CRC, reasonableness check: 13, Remote Monitoring: 3, None: 18,	control operator intervention: 3, Substitution of default value or retry: 52, Use an alternate means of Control: 4, None: 31,	Unknown Protection: 49, Shielding, anti-tamper: 13, More rigorous defect avoidance: 12, Message	1

Symptom	Control processor	Flow	Level	Pressure	Pump	VPN1	Valve	adversary	firewall1
Sensor receives bad data	27%	13%	13%	13%	0%	0%	0%	13%	20%
Sensor receives late data	43%	14%	14%	14%	0%	0%	0%	7%	7%
Actuator engages without compu	8%	4%	4%	4%	9%	39%	9%	18%	4%
Actuator Energizes incorrectly	8%	4%	4%	4%	9%	39%	9%	18%	4%
Sensor receives corrupt data	40%	0%	0%	0%	0%	0%	0%	20%	40%
Actuator Fails to Perform When	6%	5%	5%	5%	7%	47%	7%	14%	5%
Sensor receives malicious data	100%	0%	0%	0%	0%	0%	0%	0%	0%
Actuator Energizes Late	13%	6%	6%	6%	3%	47%	3%	15%	1%
Sensor receives no data	18%	18%	18%	18%	0%	0%	0%	12%	18%
Total	11%	6%	6%	6%	6%	37%	6%	15%	6%

The system effects table shows components and counts of end effects, detections, mitigations, and protections. This is an excerpt; there are a total of 24 rows in this table representing the 24 end effects that were identified in the analysis. The enables assessment of the system's dominant externally observable failure behaviors.

- **Component:** The component at the end of the propagation chain
- **System Effect:** The end effect on the system
- **Total System Effect Occurrences:** Total number of occurrences of the end effect
- **First Known Detection: Number of Occurrences:** The first detection along the propagation chain and the number of occurrences
- **First Known Mitigation: Number of Occurrences:** The first mitigation along the propagation chain and the number of occurrences
- **First Known Protection: Number of Occurrences:** The first protection and along the propagation chain and the number of occurrences
- **Severity:** Severity of the system effect

The Diagnostics table enables an assessment of what is the most likely item to have failed given the externally observable system effect. The number of rows is equal to the number of components/end effect combinations; the number of columns is the number of components (plus the adversary block). Using the top row as an example starting from the left, 27% of the failure modes that could lead to the effect of sensor receiving bad data (top data row of the table) are from the control processor, 13% are from each of the sensors themselves (flow, pressure, and level sensors), and 13% are from the adversary.

This table can be used as an aid in assessing the likely causes for a given symptom. For example, the table shows that 100% of the components that contribute to the sensor receiving malicious data are from the adversary. Hence, we call this a diagnostics table because it provides a measure of the relative likelihood of each component to be the root cause leading to the system end effect

Propagation Description Table (excerpt)

Original Failure Mode	Propagation Step 1	Propagation Step 2	Propagation Step 3	Propagation Step 4
VPN1:VPN Failure Mode: Interfered Transmissions Cause: Cyberattack	Valve:Actuator Failure Mode: Corrupt Data Cause: Unspecified Cause Detection: CRC Mitigation: Retry Comment: Protection: Unknown Protection	Pump:Actuator Failure Mode: Corrupt Data Cause: Unspecified Cause Detection: CRC Mitigation: Retry Comment: Protection: Unknown Protection	VPN1:VPN Failure Mode: Corrupt Data Cause: Unspecified Cause Detection: CRC Mitigation: Retry Comment: Requires CRC Protection: Unknown Protection	Pump:Actuator Failure Mode: Actuator Fails to Perform When Commanded Cause: Unspecified Cause Detection: Remote Monitoring Mitigation: Retry; use alternate actuation Comment: Recoverable from control station Protection: Unknown Protection Severity: 1 Severity Comment: Recoverable from control station
VPN1:VPN Failure Mode: Interfered Transmissions Cause: Cyberattack	Valve:Actuator Failure Mode: Malicious Data Cause: Unspecified Cause Detection: None Mitigation: None Comment: Protection: Message authentication	VPN1:VPN Failure Mode: Malicious Data Cause: Unspecified Cause Detection: None Mitigation: None Comment: Protection: Unknown Protection	Valve:Actuator Failure Mode: Actuator engages without computer command Cause: Unspecified Cause Detection: Unknown Detection Mitigation: Unknown Mitigation Comment: Could result in loss of control, instability, and loss of water system Protection: Unknown Protection Severity: 1 Severity Comment: Could result in loss of control, instability, and loss of water system	
VPN1:VPN Failure Mode: Interfered Transmissions Cause: Cyberattack	Valve:Actuator Failure Mode: Malicious Data Cause: Unspecified Cause Detection: None Mitigation: None Comment: Protection: Message authentication	VPN1:VPN Failure Mode: Malicious Data Cause: Unspecified Cause Detection: None Mitigation: None Comment: Protection: Unknown Protection	Valve:Actuator Failure Mode: Actuator Energizes incorrectly Cause: Unspecified Cause Detection: Remote Monitoring Mitigation: control operator intervention Comment: Could result in loss of control, instability and loss of water system Protection: Unknown Protection Severity: 1 Severity Comment: Could reuslt in loss of control, instability and loss of water system	

= Failure modes that have protections

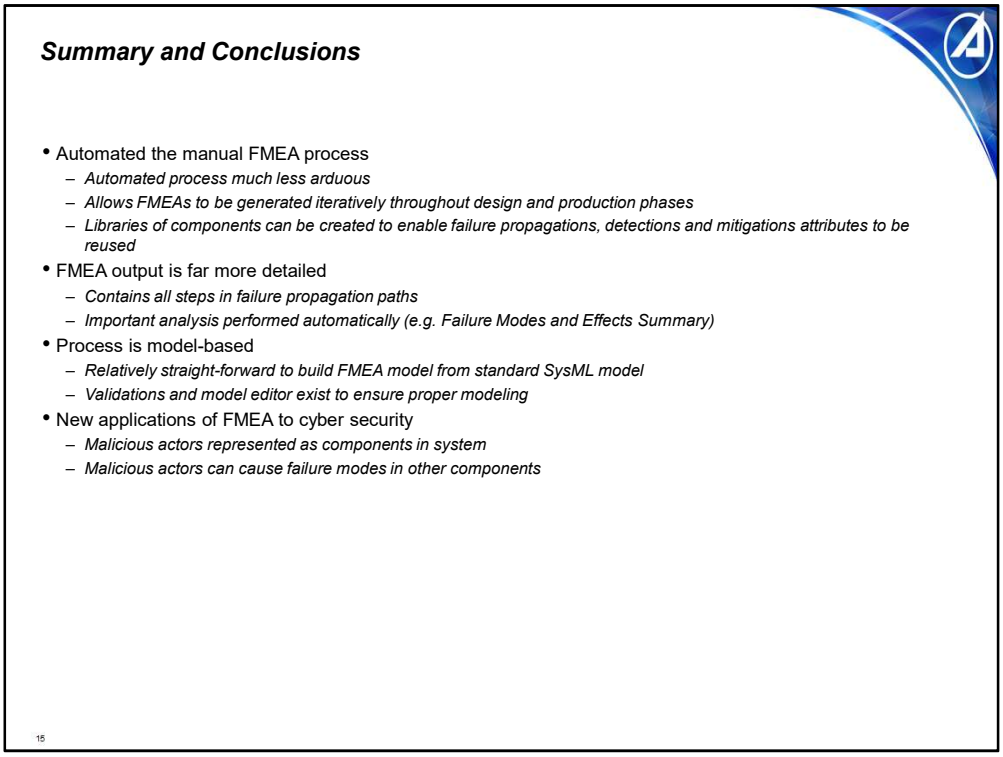
= System level effects

Shows the details of the propagation of each failure mode (expands the condensed propagation information in the Full FMEA)

14

The propagation description table shows the details of the propagation of each failure mode (expands the condensed propagation information in the Full FMEA). Each cell is a represents a single step in a failure propagation path. There are the same number of rows in the propagation as in the original FMEA (1110 in this case) The cells describe the component, failure mode, cause, detection measure, mitigation measure, a comment, and protection. To support cybersecurity assessments, propagation steps with protections are shaded in green, and the end or system level effects and shaded in brown.

A visual assessment of the state of the cybersecurity design can be readily made by zooming out of the spreadsheet. Where protection measures are absent, the analysts should examine the existing mitigation and detection measures intended to support reliability and safety and determine whether these are sufficient for cybersecurity purposes



In this presentation, we have described an automated FMEA generation capability using the SysML modeling language and described its application to a simple SCADA computer network. We also presented the outputs produced by the tool (implemented as a SysML plug-in) from this analysis and showed the insights into the design that can be achieved.

The fundamental innovation in our approach is the identification and enumeration of all failure propagation paths and the detailed documentation of the failure transformations, detection measures, mitigation measures and protective measures that can be applied to these devices to prevent or mitigate the impact of the anomaly. By doing so, we can expand the traditional FMEA approach to analysis of cyberattack vectors.

Because our approach is automated and can be readily integrated into a system development effort using Model Based Systems Engineering (MBSE), the analysis can be readily repeated throughout the design and can be used frequently to assess a system design, identify weaknesses, and take corrective actions to create a more resilient and robust system.

References

1. Schmittner C., Gruber T., Puschner P., Schoitsch E. (2014) Security Application of Failure Mode and Effect Analysis (FMEA). In: Bondavalli A., Di Giandomenico F. (eds) Computer Safety, Reliability, and Security. SAFECOMP 2014. Lecture Notes in Computer Science, vol 8666. Springer, Cham
2. Gorbenko, A., Kharchenko, V., Tarasyuk, O., Furmanov, A.: F(I)MEA-technique of web services analysis and dependability ensuring. In: Butler, M., Jones, C.B., Romanovsky, A., Troubitsyna, E. (eds.) Rigorous Development of Complex Fault-Tolerant Systems. LNCS, vol. 4157, pp. 153–167. Springer, Heidelberg (2006)
3. B. Ramanan, “An illustration of the application of Failure Modes and Effects Analysis (FMEA) techniques to the analysis of information security risks.” , August, 2008 available online at www.iso27001security.com/ISO27k_FMEA_spreadsheet_1v1.xls
4. Pierre David, Vincent Idasiak, Frederic Kratz, “Reliability study of complex physical systems using SysML”, Reliability Engineering and System Safety 95 (2010) 431–450
5. Brian Larson, John Hatcliff, Kim Fowler, and Julien Delange, “Illustrating the AADL Error Modeling Annex (v. 2) Using a Simple Safety-Critical Medical Device”, Proc. ACM 2013 High Integrity Language Technology Conference, Pittsburgh PA
6. M. Wallace, Modular Architectural Representation and Analysis of Fault Propagation and Transformation, Proc. European Joint Conf. Theory and Practice of Software (ETAPS), Elsevier Electronic Notes in Theoretical Computer Science(ENTCS),vol.141,no.3,2005,pp.53–71.