2021 NDIA GROUND VEHICLE SYSTEM ENGINEERING AND TECHNOLOGY SYMPOSIUM
CYBER TECHNICAL SESSION
AUGUST 10-12, 2021 - NOVI, MICHIGAN

SECURE RAPID PROTOTYPING FOR UNMANNED SYSTEMS

Hal Aldridge, PhD[1], **Fred Livingston, PhD**[1]

[1]Secmation, Raleigh, NC

ABSTRACT

Considering the growth of unmanned vehicles in Defense and Government applications, a simple and efficient way to design, develop and deploy trusted and secure systems is imperative. Secmation's SecMUAS brings a platform for the rapid design and development of secure modular unmanned systems to defense applications and beyond. SecMUAS "bakes in" cybersecurity features using a modular design framework for unmanned systems. SecMUAS enables affordable, high assurance, "future-proof" solutions to rapidly transition from design to operational use. Secmation's SecMUAS hardware and software will provide developers a capability to address cybersecurity requirements and related certification approval processes, enabling the rapid transition of technology to the warfighter.

Citation: H. Aldridge, F. Livingston, "Secure Rapid Prototyping for Unmanned Systems", In *Proceedings of the Ground Vehicle Systems Engineering and Technology Symposium* (GVSETS), NDIA, Novi, MI, Aug. 10-12, 2021.

1. INTRODUCTION

Unmanned Systems have become an essential element of US military operations [1]. The operational users have a growing need for customized vehicles to meet dynamic battlefield requirements for combat support and intelligence gathering. Secmation is developing SecMUAS (Secure Modular Unmanned Aerial Systems) for the Office of Naval Research to address security and customization requirements. While the initial application is UAS, the system is designed to support unmanned ground, surface, underwater, and space applications. SecMUAS promises to become a foundational element of DevSecOps for unmanned systems. This paper will provide a summary of the SecMUAS solution.

2. SECMUAS OVERVIEW

SecMUAS provides an advanced security architecture and intuitive user interface that enables designers to design a secure unmanned system without security expertise. The SecMUAS design process is summarized in Figure 1. The SecMUAS Configuration Integrated Development Environment

DISTRIBUTION STATEMENT A. Approved for public release: distribution unlimited.

(IDE) provides a rapid, unmanned system software development suite that automatically incorporates security features to implement a security policy. The Configuration IDE enables the user to manage the SecMUAS design and build process. Through the IDE, the user can select which software and hardware elements they need to construct an unmanned system with the desired characteristics. The design process includes selecting a security policy that the Configuration IDE will enforce in constructing the vehicle design. As part of the vehicle configuration process, the Configuration IDE will produce documentation on how the design meets the selected security policy. This documentation can be used as part of a security approval process (e.g., DoD RMF) for the unmanned system. SecMUAS incorporates a US-designed and manufactured Secure Control Unit with advanced security and performance features providing a secure root of trust. SecMUAS also supports a library of validated unmanned systems hardware/software components, an NSA-certified communication system, and a ground station, enabling complete unmanned system design and integration.

3. SECURE RAPID PROTOTYPING

3.1. Configuration IDE

The designer will interact with SecMUAS using the Configuration IDE. The Configuration IDE will enable the designer to configure parameters such as software components, security levels/policies, and hardware components to be used and enforced by design. By interacting with a Graphical User Interface (GUI), the designer can quickly create or modify designs enabling rapid prototyping.

Key features of the Configuration IDE include security policy, rules engine, and hardware/software (HW/SW) database.

The security policy contains the technical security controls that the unmanned system must meet and would be designed to align with the certification/approval process. Cybersecurity experts may initially provide this policy within the designer's organization. This policy would feed into the rules engine. The rules engine is a critical component in the Configuration IDE. The rules engine uses the security policy, the unmanned system configuration input by the designer, and the metadata information in the HW/SW database on the components selected to determine if the resulting design can meet the security requirements. If the design does not meet security requirements, the Configuration IDE will attempt to update the configuration to meet the security policy automatically.

The rules engine verification can result in 3 states: success, recommendation, or failure. On success, the Configuration IDE was able to update the system configuration to meet policy. On recommendation, the Configuration IDE identifies options that would enable the configuration to meet policy but require the designer to review non-security-related issues. On failure, the Configuration IDE does not find a configuration that meets policy and generates a list of the policy elements it could not address.

While the designer has final approval of all SecMUAS generated designs, an example for the recommendation state is as follows. The security policy requires data transmitted from the unmanned system to the base station to be encrypted using cryptography with a specific security certification. The designer has not selected a radio system with the required encryption capability. The Configuration IDE finds any radio systems in the HW/SW database that satisfy the security policy and suggests them to the designer. The metadata related to the component type (radio) requires designer approval before incorporating it into the configuration. The radio may have different operating characteristics (e.g., frequency, bandwidth) than the radio initially selected. The Configuration IDE communicates the recommendation to the designer for review and approval.

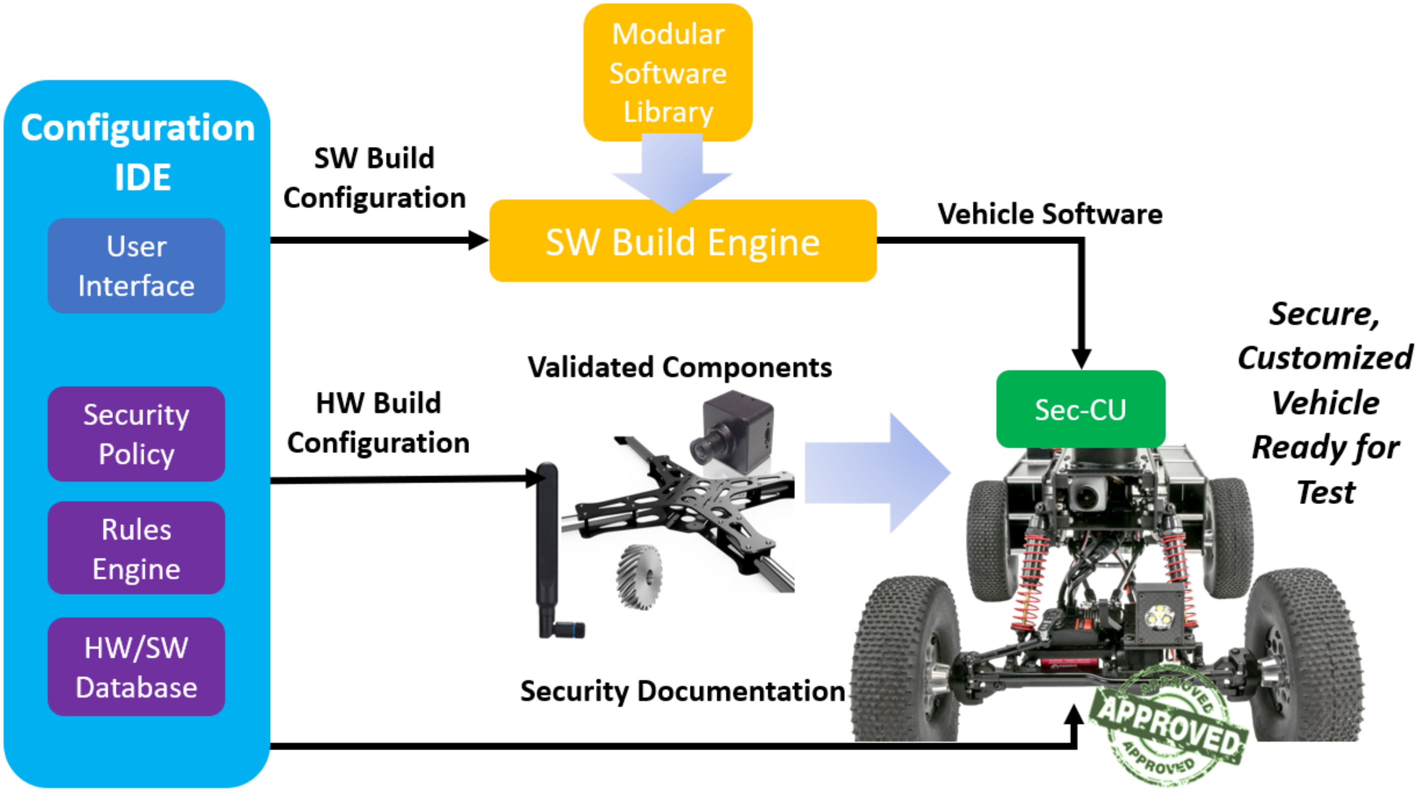

Figure 1: Sec-MUAS System Overview.

3.2. Build System

The SecMUAS build system translates the design validated by the Configuration IDE into executable software, configuration files, and security elements to provide the unmanned system. The system leverages modular software components and off-the-shelf software configuration tools to build the software.

3.3. Modular Software Library

The use of modular software components in SecMUAS will enable flexible and rapid prototyping while meeting a defined security policy. This is accomplished through two fundamental mechanisms. The first is the Modular Software Library (MSL). The MSL contains software modules that implement different system functions that have been validated/approved for use in unmanned systems software configurations. The modules include well-defined interfaces (e.g., I/O data types, number of I/O ports), enabling them to be connected to implement a helpful software configuration. This model is similar to that found in Simulink, GNU Radio, and other graphic languages.

Software within the MSL will be tied to metadata which defines its use in different configurations. For example, if an encryption service is defined as approved to be used as part of a Medium level security policy, it would not be allowed to be included in a system where the security policy is defined to be High. Similarly, suppose a system is targeted at Operational use. In that case, the configuration rules enforced by the Configuration IDE might be set not to allow modules marked Experimental to be used in the software configuration without proper authorization.

The template for a connection between components would be initially selected from a standard library and eventually generated using the Configuration IDE as SecMUAS matures. This template would be selected and/or generated to be consistent with other security policies, build parameters, and loading parameters generated by the Configuration IDE. The software designer would use the selected and/or generated template to integrate their existing code with the SecMUAS architecture. Once the template is customized and the existing software integrated and/or tested with the custom template, the resulting modular component can be added to the software module library and re-used in different unmanned system configurations. This process allows for fast integration of robotics algorithms, such as from the ROS-M community.

3.4. Hardware Library

Many small unmanned systems used by the DoD are based on COTS components to enable rapid capability deployment and reduce cost. A key element to building a secure system is to have a validated supply chain. COTS components, especially "hobbyist" grade components, can vary widely in quality and performance. In addition, depending on the type of component, these COTS components can have malicious software, software vulnerabilities, or other issues that will result in a security risk for the unmanned system. This validated component list will form the modular hardware list that the designer can select using the Configuration IDE to configure the unmanned system. The component types evaluated include Actuators, Motor Controllers, Motors, Sensors, Cameras, Drive trains, and Power systems. SecMUAS also incorporated into the library of validated hardware components, an NSA-certified communication system, and a ground station with an operational pedigree, enabling complete unmanned system design and integration. The SecMUAS provides a guide for importing additional hardware components to its ecosystem using the Configuration IDE.

3.5. Security Documentation

As part of the vehicle configuration process, the Configuration IDE will produce documentation on how the design meets the selected security policy. This documentation can be used as part of a security approval process (e.g., DoD RMF) for the unmanned system. The key element of the documentation generation process is utilizing the artifacts generated by the Configuration IDE. To validate the security policy, the Configuration IDE has both the details of the security policy (e.g., the requirements) and the controls used to address these requirements. This information is used along with automated document generation tools to produce a document detailing the compliance to the security policy and compliance methods. It is envisioned that the security policy would align with the security approval process. The availability of automatically generated documentation that aligns with the "as-built" security configuration promises to significantly reduce the labor and time required to prepare for and obtain security approval.

3.6. Secure Control Unit (Sec-CU)

Once a system is configured, a software image must be built to be loaded and executed on the vehicle. The software must be loaded onto a computer system with a robust supply chain pedigree to remain secure. Without this pedigree, the computer system may compromise the security of the unmanned system software through vulnerable and/or malicious hardware and software from the manufacturer. The SecMUAS program is developing a Secure Control Unit (Sec-CU) to provide a secure root of trust and core computing platform. The Sec-CU is built around System on Module (SoM). The SoM is a small computer board of similar size to a credit card. It contains the main computing element, the System on Chip (SoC), and support electronics to provide the computing capability for the Sec-CU. The use of an SoM enables the controller to keep up with rapidly evolving chip technologies. The SoC is selected to have essential hardware-enforced security properties enabling the SecMUAS security architecture.

The SoM will be attached to a custom carrier board designed to meet the needs of multiple sizes/types of unmanned systems. The carrier board will provide I/O, power control, and other support functions but no computing elements. The carrier board will also offer a removable memory component (e.g., SD Card or similar), enabling mission data to be saved for later analysis. The Sec-CU will connect to vehicle, payload, and communications systems through different interface types, allowing various modular components to be incorporated in system design. At the core of the hardware, the security architecture is a COTS System on Chip (SoC) component.

Modern SoCs are complex devices that incorporate multiple processing cores, peripherals, and security features in one integrated device. The selection of the SoC took into account computational capability, I/O needed to operate an unmanned system, security features, and SWAP (Size, Weight, and Power), among other considerations. The NXP i.MX8 family of applications processors was selected, with the baseline being the i.MX8X component. In addition to the computing power provided by the A35 cores, the i.MX8X also includes ARM NEON DSPs, a GPU, video encoding/decoding, and multiple peripherals needed for vehicle control.

4. CONCLUSION AND FUTURE WORK

The SecMUAS solution provides an expandable rapid prototyping framework built to support secure unmanned systems development. This framework will enable the rapid transition of capabilities to the warfighter by reducing development and certification timelines.

5. ACKNOWLEDGEMENT

This material is based upon work supported by the Office of Naval Research under Contract No. N68335-21-C-0150.

References

[1] K. Snyder, N. Trenaman, and J. Coffey, "Development and Deployment of Multi-mission Intelligent Autonomous Maritime Systems", in *Global Oceans 2020: Singapore – U.S. Gulf Co*, 2020, ISBN: 9781728154466.