Systematic Analysis of Cyber-Attacks on CPS – Evaluating Applicability of DFD-based Approach

Mark Yampolskiy, Peter Horvath, Xenofon D. Koutsoukos, Yuan Xue, Janos Sztipanovits
Vanderbilt University, Institute for Software Integrated Systems (ISIS)
{mark.yampolskiy, peter.horvath, xenofon.koutsoukos, yuan.xue, janos.sztipanovits}@vanderbilt.edu

***Abstract*—Cyber-Physical Systems (CPSs) consist of as well as interact with cyber and physical elements. This creates multiple vectors for CPS-internal (i.e., within CPS) as well as for CPS-external (i.e., between CPS itself and its environment) Cyber-Physical Attacks. We argue that an effective Cyber-Physical Defense can only be elaborated if possible attacks on CPS can be identified and assessed in a systematic manner. In this paper, we focus on cyber-attacks only. Our contribution in this paper is the following. We assess the applicability of Data Flow Diagrams (DFD) for the systematic analysis of cyber-attacks against CPS. In this context, we introduce several extensions to DFD. We evaluate the analysis procedure by applying it on a comparatively simple example of a quad-rotor UAV. The selected UAV is fully functioning and contains multiple structural elements representative for more complex systems. At the same time, its simplicity enables an in-depth manual analysis. Our analysis shows that cyber-attacks executed against CPS can lead to various cyber-physical interactions. This, in turn, creates novel challenges for CPS defense. Finally, we outline the preliminary results of our work towards a Taxonomy of Cyber-Physical Attacks.**

***Index Terms*—Cyber-Physical Systems (CPS), CPS security, Cyber-Physical Attacks, CPS vulnerability assessment, Taxonomy.**

I. Introduction

The term *Cyber-Physical System* (CPS) is commonly defined as a system with the following three capabilities: (i) sensing physical world (e.g., the distance to the car ahead), (ii) making decisions (e.g., whether it is necessary to decelerate), and (iii) performing actions in physical world (e.g., activate brakes). CPSs become increasingly internetworked with the cyber world, including communication with control stations, other CPSs, or even the Internet. This exposes CPSs to various kinds of cyber-attacks.

At the same time, the amount and the diversity of CPS embedded in our daily life steadily grows. Just to name a few examples, diverse elements of modern cars and airplanes, all kinds of unmanned vehicles, smart homes, smart cities, and various critical infrastructures. This all makes us dependable on CPSs functioning properly and therefore vulnerable to CPSs failure or faulty behavior.

The importance of CPSs in our daily life and their vulnerabilities to cyber-attacks make such systems a very attractive target. The quantity and frequency of such attacks grows steadily. At the same time, the motivation of adversaries, complexity of performed attacks, and the attack consequences differ among known incidents. For instance, in 2000 the Maroochy Shire Council's sewage control system in Queensland, Australia, caused the flooding of its nearby surroundings [1] [2]. This attack was performed by a fired employee as an attempt to enforce his rehiring. Another prominent example is the Stuxnet, which has led to the physical damage of centrifuges at an Iranian uranium enrichment plant while hiding the attack behind the previously recorded status data [3] [4]. A good overview of several further examples can be found in [5].

We argue that our ever growing dependability on CPSs requires that these systems become resilient against *Cyber-* and *Cyber-Physical Attacks*. We further argue that the elaboration of an effective CPS defense requires that the potential attacks on CPS can be discovered and assessed in a systematic manner. Despite the urgency of this topic, the current public research is dominated either by practical examples highlighting the vulnerability of selected systems against attacks, e.g., how electronics of a modern car can be compromised [6], or by very general descriptions of possible approaches and relevant research areas, such as classification of intruder entry points and cyber consequences [7].

The "missing link" in the existing work is a systematic procedure, which can be applied to a wide range of CPS in order to assess their vulnerability. Therefore, we see the necessity to evaluate the applicability of security vulnerability analysis approaches established in computing systems and networks on CPS.

Our contribution in this paper is the following. We assess the applicability of Data Flow Diagrams (DFD) for the systematic analysis of cyber-attacks against CPS. We introduce several extensions to DFD necessary to reflect physical and cyber-physical interaction in CPS. On the selected example, we illustrate how Extended DFD (xDFD) can be used for the systematical CPS vulnerability assessment. Based on the case study results, we discuss important properties of attacks on CPS, such as frequent "outbreaks" from cyber domain. We present preliminary results of our work on the *Cyber-Physical Attack Taxonomy*, which should capture the cross-domain and cross-layer nature of Cyber-Physical Attacks on CPS.

This paper is structured as follows. In Section II, we give an overview of related stated of the art. In Section III, we describe the architecture of a quad-rotor UAV – a CPS we have selected for the evaluation purposes. In Section IV, we analyze the applicability of Data Flow Diagrams (DFD) for the systematic analysis of cyber-attacks on CPS. First, we introduce extensions to DFD, which are needed in order to

model the selected UAV. Then we present how the selected UAV can be modeled with the extended DFD. Later in this section, we illustrate how Extended DFD can be used for the systematic analysis of possible cyber-attacks. Section V presents the preliminary results of our work on Cyber-Physical Attack Taxonomy. A short discussion about lessons learned and an outline of our future plans conclude this paper.

II. Related Work

Several research areas are relevant for our present work. Below we present works belonging to following categories: dependability and security, CPS security, and taxonomies of cyber-attacks.

A. Dependability and Security

Dependability and security are both essential properties for a CPS. Dependability was defined in [8] as a global concept incorporating multiple quantitative parameters. In its latest version [9], dependability taxonomy incorporates five properties: availability, reliability, safety, integrity, and maintainability. It also contains an attempt to incorporate some security related properties (availability, confidentiality, and integrity) as a part of the taxonomy. Security is often considered from the angle of information security with the following six properties: availability, confidentiality, integrity, authentication, authorization, and nonrepudiation [10]. Thus, dependability and security concepts are derived from partly overlapping properties. In [11], discussing both topics, authors point out the principal difference between dependability and security approaches. Whereas in the dependability community it is common to assume stochastic models for element failures, for the security evaluation it is in general not valid. Both dependability and security have been elaborated as concepts for network and computing systems.

B. CPS Security

The fact that CPS are vulnerable to cyber-attacks was recognized as an issue only a few years ago. The necessity of cyber-defense against such attacks is frequently motivated by the fact that "most uses of cyber-physical systems are safety-critical" [12]. Currently, the majority of CPS-security related works are focusing on critical infrastructure.

An interesting overview of various cyber-attacks on critical infrastructure can be found in [5]. A comprehensive analysis of critical infrastructure is given in [13]. In [13], authors argue that it is essential to understand interdependencies between various critical infrastructures. They identify four classes of such interdependencies: physical, cyber, geographical, and logical. However, not all of these classes are applicable to CPS of our concern – such as remote controlled UAVs. In the more recent work [14] the author stresses the necessity to model and to simulate the critical infrastructure. These techniques should allow identification of effects in the case of component failure. We think that such approach is generally applicable to all kinds of CPS. However, simulation approaches greatly depend on the understanding of interdependencies between elements, detail grade of the model, and its compliance with the real system.

In [7] authors consider Smart Grid Infrastructure. They give a well-structured high-level overview of cyber security requirements, attack models, and possible countermeasures for a given system. However, it remains unclear which systematic approach authors have used in order to elaborate the presented results. Therefore it is also unclear whether this approach can be applied to other CPS kinds too.

Regarding vehicle-like CPS, the current public research is dominated by the practical examples showing vulnerabilities of selected systems. Especially interesting and relevant for the current paper is the contribution in [6]. The authors present an impressive sequence of cyber-attacks executed on the modern car's electronics. Even though authors present the experimental methodology, they don't give any systematic approach which could be applied to assess the vulnerabilities of any other CPS. There are several other works falling in the above mentioned category. However, we are not aware of works pursuing systematic approach and at the same time mature enough to be applicable on the real systems.

A very interesting discussion about differences between computer systems and CPS is given in [15]. The authors point out that various timing aspects (e.g., for communication between components) are intrinsic for CPS. Those aspects are very well known in the area of embedded programming. However, these aspects are usually neglected in design and analysis of software at higher layers. Unfortunately, this also includes all kinds of security analysis of network and computing systems.

The analysis of cyber-attacks in computer systems and networks has a very long history and contains multiple approaches. For our current investigation, we have selected Data Flow Diagrams (DFD) [16] [17] based approach. This approach enables a systematic analysis cyber-attack based on the model of the software components and their interactions.

C. Taxonomies of Cyber-Attacks

We are aware of only a single proposal for the taxonomy covering both cyber and physical aspects natively. In [13], authors focus on critical infrastructures, more precisely on interactions and interdependencies between such infrastructures. Authors give a very comprehensive analysis of such interdependencies and present a corresponding taxonomy. Authors propose six dimensions: environment, coupling and response behavior, type of failure, infrastructure characteristics, state of operation, and type of interdependencies. Authors argue that it is essential to understand interdependencies between various critical infrastructures. We see a significant similarity to our position, as we think that it is essential to understand the CPS structure as well as the interdependencies between involved cyber and physical components.

In networked and computing systems taxonomies have been used for both a post-factum attack classification and for detection of attack possibilities. Several criteria for taxonomy have been elaborated, such as unambiguity, completeness,

or mutual exclusiveness. A very good summary of various taxonomy related requirements can be found in [18]. However, as pointed out by authors, not all taxonomies should fulfill every listed criterion. For instance, not all taxonomies strive to be mutually exclusive. In [18], authors discuss characteristics of cyber-attacks and conclude that a tree-like taxonomy is not well suitable for the description. Instead they propose classification based on four dimensions: (i) attack vector (i.e., method by which an attack reaches the target), (ii) attack target, (iii) exploited vulnerability, and (iv) additional payload or effect beyond the attack themselves. The purpose of the proposed taxonomy is the classification of detected attacks.

Further, we would like to mention taxonomy proposed in [19]. In this proposal, the classification is performed based on five dimensions: Attack Vector, Operational Impact, Defense, Information Impact, and Target. Especially interesting is the Defense dimension. We see that possible countermeasures should be considered and captured along with the possible attacks.

III. Analyzed Platform: AscTec Hummingbird

For our case study we have selected AscTec Hummingbird quad-rotor UAV [20]. It suits perfectly our study for two reasons: (i) it is simple enough to perform manual in-depth analysis, and (ii) it contains structural elements characteristic for the more complex systems, such as multiple processors, bus communication, and isolated communication segments.

The AscTec Hummingbird consists of a core and four booms. Almost all electronics as well as the battery are located within the core. The four motors and propellers are mounted at the ends of the booms. The four motor controllers are located on the booms too.

The controlling electronic of the Hummingbird consists of two control units, referred to as High-Level (HL) and Low-Level (LL) processors, and four specialized motor controllers (see Figure 1). The HL processor is reprogramable by the owner of the Hummingbird. The SDK available for download from the vendor website provides a good starting point for the development of user-specific applications. In order to burn new software into the HL flash, it should be booted with two dedicated pins physically connected, i.e., overwriting of flash is physically protected and requires physical access to the hardware. For our analysis, we will also consider the case when it is not needed, i.e., it is possible to overwrite the code without physical access to the device. The LL processor code is encrypted and not intended to be modified by the user.

Internal communication between HL and LL processors is performed via Synchronous Serial Port (SSP) bus. There is neither physical nor direct logical connection between HL processor and the four motor controllers. Instead, the HL processor passes all commands to the LL processor. The LL processor interprets these commands and passes the sequence of corresponding commands to the motor controllers.

The UAV's communication with the external world is realized in both cyber and physical domains. In cyber domain, both HL and LL processors can communicate with the remote

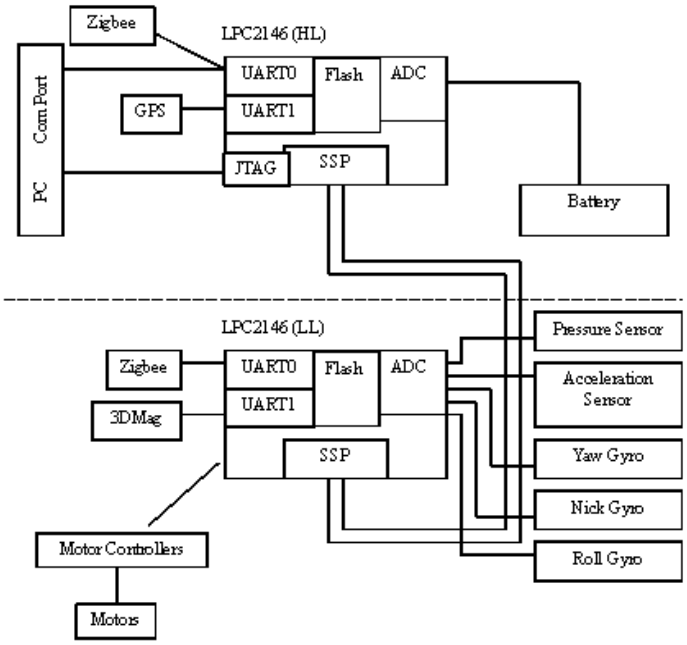


Fig. 1. AscTec Hummingbird, Hardware Architecture

control (R/C) and/or computer via Zigbee wireless interface. In our investigation, we consider three possible hardware variations: (i) only the HL processor has an access to the Zigbee module (ii) both HL and LL can communicate with the same Zigbee module via a shared bus, and (iii) both processors have access to two separate communication modules.

Under physical communication we understand several LEDs and beeper, which can produce signals to notify the operator about various events. For instance, the beeper is used in the case if the battery charge level is becoming too low. This is especially important because a Lithium-Polymer battery is used, which (according to the specification) can be irreversible damaged or even explode if it discharges below 9 Volt during the flight. Also here we investigate three cases: (i) only the HL processor has access to the LEDs and the beeper, (ii) both processors have access to the same LEDs and the beeper, and (iii) both processors have access to different signaling devices.

The above mentioned description reflects interactions during the operation of the quad-rotor UAV. In order to reflect all possible interactions, we should also consider interactions between UAV and a computer during the maintenance phase. In order to upload and to burn new flash code to the AscTec Hummingbird's HL processor, it has to be booted when two dedicated pins are physically interconnected. We again consider two cases: (i) such boot approach is necessary and the code execution of HL processor is halted, and (ii) such boot approach is not necessary and the code of HL processor can be executed during the connection with the external computer. The second mode is especially interesting because in more complex systems like cars it might be more commonly used, e.g., for accessing of monitoring data.

IV. DFD-Based Approach for Threat Analysis

In this section we first introduce Data Flow Diagrams (DFD) and then propose several extensions to it. We model the selected UAV with the Extended DFD (xDFD). We present how this model can be used for the systematic analysis of possible cyber-attacks. In order to highlight the cross-domain and cross-layer nature of the attack, we present several selected attacks discovered with the proposed approach.

A. Modeling with Extended DFD (xDFD)

The *Data Flow Diagrams* (DFD) are commonly used for the threat analysis of software systems [16] [17]. The DFD elements and corresponding symbols are listed in Table I.

Item	Symbol
Data flow	One way arrow
Data store	Two parallel horizontal lines
Process	Circle
Multi-process	Two concentric circles
Interactors	Rectangle
Trust boundary	Dotted line

TABLE I
DFD Symbols (according to [17])

Unfortunately, existing DFD elements are insufficient for the full-fledged description of all CPS-relevant interactions. For instance, it is impossible to distinguish between cyber and physical communications. Therefore, we propose several extensions to DFD. These extensions should suit following purposes: (i) enable description of the physical elements along with the cyber ones, (ii) enable description of the physical data flow along with the cyber one, and (iii) enable description of the communication medium along with the communication flow realized upon this medium. The additional elements we have introduced as a part of the *Extended DFD* (xDFD) are presented in Table II.

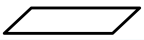
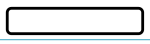
Item	Symbol
Physical component	
Communication medium	
Optional data flow	
Physical signal	

TABLE II
Extensions for DFD symbols

Please note that these are extensions needed for the cyber-attack analysis on the selected UAV only. Therefore, we see our proposal as a first step for the extension of DFD symbols. Support of more complex CPS might require further

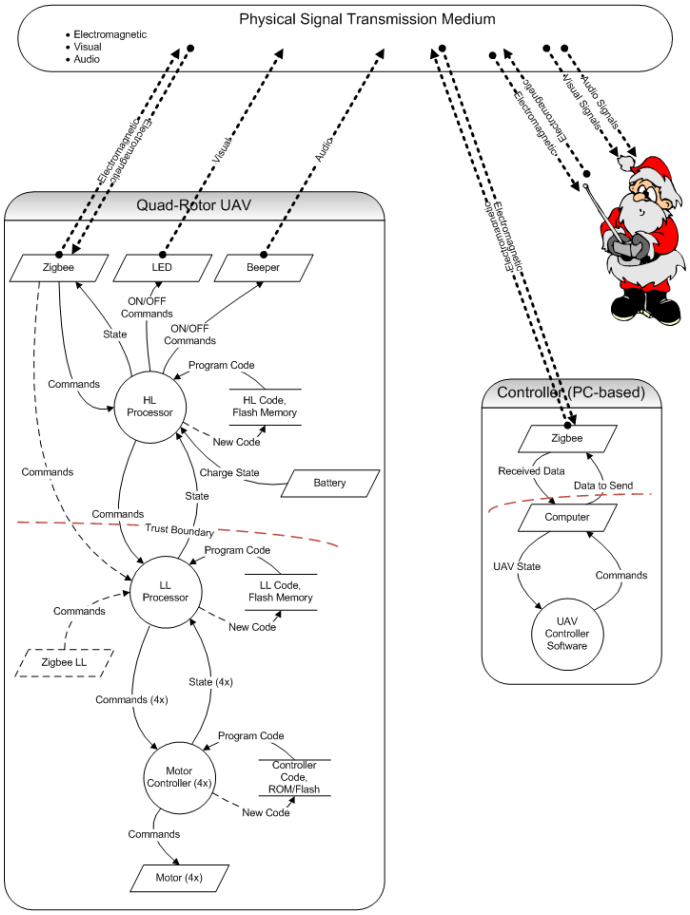


Fig. 2. Extended DFD Diagram, Operation of UAV

extensions, e.g., for the description of interdependencies between processes and hardware infrastructure they (commonly) use. Support of the physical attack analysis might require support of further elements in order to represent the physical components, their interdependencies, as well as the physical interactions with the environment. For instance, the heat and electromagnetic dissipation of CPS elements and their influence of the other elements proper behavior might be required for the analysis of physical effect propagations in Cyber-Physical Attacks.

B. Extended DFD based Approach

The extended DFD diagrams for the selected UAV during its operation and maintenance phases are depicted in Figures 2 and 3 respectively. From the architectural point of view, the chosen CPS use case combines elements of computing and networking systems, as well as extends those into the physical domain. Note that also CPS-internal interactions between different components such as the HL and the LL processors can be seen as a networked system and therefore exposed to the corresponding threats and adversarys goals.

The systematic approach based on the extended DFD can be seen as a ”walk” through the diagram elements. For every cyber element in our xDFD diagrams, we have analyzed which

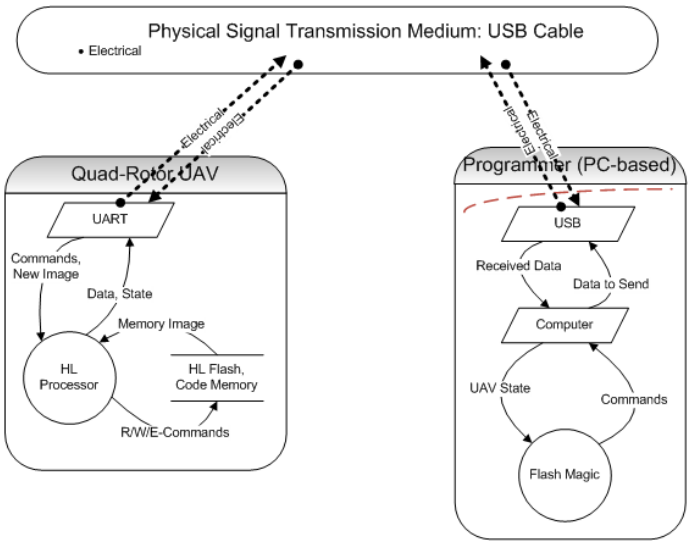


Fig. 3. Extended DFD Diagram, Maintenance of UAV

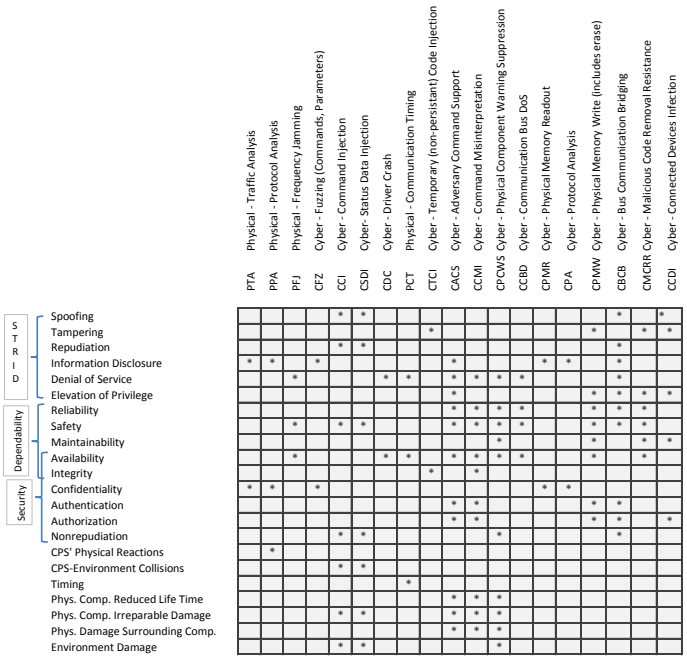


TABLE III
IDENTIFIED ATTACKS

cyber-attacks on these elements or from these elements are possible. We have categorized attacks based on the STRIDE threat model. STRIDE is an acronym for Spoofing, Tampering, Repudiation, Information Disclosure, Denial of Service, and Elevation of Privilege [21]. We were especially cautious about how an attack can affect dependability (i.e., availability, reliability, safety, integrity, and maintainability) and security (i.e., availability, confidentiality, integrity, authentication, authorization, and nonrepudiation) properties. We have also paid special attention to the potential effects of an attack which exceed the violation of the above mentioned properties. In total, we have identified 19 principally different types of cyber-attacks on the selected UAV. Please note that we have abstained from counting all possible attack variations as well as from counting the same attacks on different elements. Those attacks are summarized in Table III. Valid intersections between attacks and STRIDE as well as dependability or security properties are marked with "*". This table further contains several identified "outbreaks" into the physical world.

In the presented work, our main goal is to evaluate the applicability of DFD-based for the CPS vulnerability assessment. Therefore, below we present only a selection of the identified attacks, which emphasize complex cyber-physical interactions. All described attacks are grouped based on the element in the diagram which they affect and/or use. For every attack we provide its textual description. In order to distinguish between attacks, we identify each attack with an acronym based on the name we give an attack. We further specify the attack's STRIDE effect as well as which dependability and security properties the attack violates or is able to violate. The "Other" section of the attack description presents aspects not covered by the classical cyber-security. Please note that the identified attacks can violate multiple properties at the same time.

1) Wireless Communication between UAV and Controller: As communication between the UAV and either the Remote Control (R/C) or the controlling computer is performed via wireless connection, it is exposed to several attacks.

Attack ID: PPA (Physical - Protocol Analysis)
Description: The used protocol can be analyzed based on the correlation between eavesdropped messages (belonging to the cyber domain) with the observations of the UAV's physical reactions to these messages, e.g., increasing/decreasing of thrust, rotation, or shutting down the engine.
Defense: Countermeasures are possible in both cyber and physical domains. In cyber domain, a simple encryption of the sent commands does not really provide any protection from the protocol analysis and the reply attacks because in this case the adversary can correlate with the physical reaction on the encrypted command. Usage of cypher block chaining encryption modes can improve the robustness of the protocol against the traffic analysis. The well-established countermeasure against various kinds of record-[modify]-reply-attacks is usage of message numbers, as well as of the cryptographic checksums. Those, however, come at costs of reduced bandwidth utilization and increased CPU consumption. In physical domain, frequency hopping can increase the robusteness of the protocol to the analysis.
STRIDE-Type: Information Disclosure
Violation of Dependability: N/A
Violation of Security: Confidentiality
Other: Can use CPS' Physical Reactions for analysis

Given that the wireless communication protocol is compromised, an adversary can start cyber-attacks either against the

UAV or against the controller.

Attack ID: CCI (Cyber - Command Injection) and CSI (Cyber- Status Data Injection)

Description: In [22], authors distinguish between attacks against estimation and control algorithms. Similarly, in our example an attack on the communication link from the controller to the UAV can be seen as an attack against control; an attack against the communication link in the opposite direction can be seen as an attack against estimation. The latter is especially important if an UAV is controlled beyond the line of sight[1]. Both the wrong estimation of the UAV position as well as the injection/reply of malicious commands can lead to the situation where the UAV leaves the designated area or even collides with an obstacle. Such collision can lead to the physical damage of the UAV, its environment, or even injuries of persons. Also without any collision between UAV and its environment, this attack can lead to physical consequences, e.g., hovering in the middle of the road can endanger the traffic situation and lead to a car crash. All this can lead to legal consequences or even inflame social discussion about the acceptance of UAVs.

Defense: A common defense measure is the combination of a message number and the cryptographic checksum. This comes at the cost of the reduced bandwidth utilization and increase of the computation.

STRIDE-Type: Spoofing, Repudiation

Violation of Dependability: Safety

Violation of Security: Nonrepudiation

Other: CPS-Environment Collisions, Physical Component Irreparable Damage, Environment Damage

2) HL Processor: Once the malicious code is injected into the HL processor, e.g., via the buffer overflow[2], a broad multitude of attacks is possible. As the HL processor is dedicated to define the application objectives, the malicious code can[3] accept adversary command which are not a part of the defined application layer protocol, interpret the user-commands in an inappropriate for the end-user unpredictable way, send wrong/modified status data to the controller, send data to the adversary, enable or disable LEDs and the beeper, ignore/misinterpret the battery charge state, send wrong/modified commands to the LL processor, and finally eavesdrop or disrupt communication between devices connected to the same bus. This means that the code injection into the HL processor can have very severe consequences. Below we discuss one of the above mentioned possible attacks staging from the HL processor in more details.

Attack ID: CPCWS (Cyber - Physical Component Warning Suppression).

Description: The UAV uses Lithium-Polymer battery which should not be discharged below 9 volt. Compromised HL processor can, for instance, send false or slightly modified battery state to the controller and at the same time prevent acoustic warning signal. The consequences can have three severity levels: (i) reduced battery lifetime, (ii) irreparable damage of the battery, and even (iii) physical damage of the surrounding components. We would like to highlight that in general such consequences are not limited to the battery only, but can be applied to various cyber-controlled physical unit.

Defense: Redundant sensors as well as sensing data evaluation devices can significantly improve the robustness against such attack. Also signaling possibilities should be redundant.

STRIDE-Type: Denial of Service

Violation of Dependability: Reliability, Safety, Maintainability, Availability

Violation of Security: Availability, Nonrepudiation

Other: Physical Component Reduced Life Time / Irreparable Damage, Physical Damage Surrounding Components, Environment Damage

3) USB Communication between UAV and PC during Maintenance: Finally we would like to point out that the adversary's attacks might target not (or not only) the CPS itself but rather the computer which will be connected to the UAV during its maintenance.

Attack ID: CCDI (Cyber - Connected Devices Infection)

Description: Currently, it is common to connect various external devices via an USB interface. Even though the AscTec Hummingbird is supposed to be booted in the special mode for the flash read/write/erase operations, it might happen that the cable is connected during the UAV's HL processor code is running in normal mode and the malicious code is active. This enables all kinds of attacks via USB connection which became very common in the recent years. More complex systems which don't require such boot sequence are even more vulnerable to this kind of attack.

Defense: Automatically stop of the code execution at the CPS in the case if an external device is connected might appear to be a good idea. However, we would like to dissuade the reader from this point of view. The reason is that it can be misused by an adversary, who can gain physical access to the CPS. Therefore, the only viable option is the presence of firewall and antivirus software on the maintenance computer.

STRIDE-Type: Tampering, Elevation of Privileges

Violation of Dependability: Maintainability

Violation of Security: Authorization

Other: N/A

[1]In the case of AscTec Hummingbird, documentation prescribes that the UAV should remain under the operator's personal surveillance during the whole flight time.

[2]For a comprehensive description of possible vulnerabilities in code as well as the methods to avoid them, we would like to point the reader to [21].

[3]Here we enumerate the possibilities "going" clockwise in DFD diagram, beginning with the connection to Zigbee.

V. Towards Taxonomy of Cyber-Physical Attacks, Preliminary Results

Taxonomies are commonly used for either the classification of already detected or the in-advance identification of possible attacks. Analyzing identified attacks, we are currently working on the taxonomy serving both these purposes. Below we outline the preliminary results of our work on the *Cyber-Physical Attack Taxonomy*.

We see that the biggest challenge of structuring Cyber-Physical Attacks into a single taxonomy originates from the structural heterogeneity among different attacks. This means that different attacks have to be characterized by varying (i.e., not always the same!) combinations of different classification dimensions; the usage of other dimensions might be either irrelevant or implicitly clear based on the attack.

Nevertheless, analyzing identified attacks, we have identified that every attack on CPS contains two distinct elements either implicitly or explicitly: (i) element(s) which is/are *targeted* (or *influenced*) by the attack and (ii) the *victim* element(s) the change of which state(s) defines the *effect* of the attack. Both these elements can be located either in the cyber or in the physical domains. Therefore, we refer to these dimensions as "*Target Domain / Influenced Element*" and "*Effect Domain / Victim Element*" respectively. For the identified attacks, below we enumerate elements of both dimensions in a structured (tree-like) manner.

Target Domain / Influenced Element

- Cyber
  - Communication Channel (CPS $\Leftrightarrow$ R/C)
    - Communication Protocol
    - Command/Data Exchange
    - Delay
    - Jitter
  - CPS
    - Network Driver
    - Bus Communication
    - Controlled Plant Algorithm (CPS)
    - Executable Code (Process)
    - Flash Memory
  - Remote Controller / Computer
    - Estimation Algorithm (Controller)
    - Operating System
  - Maintenance Computer
- Physical
  - Communication Channel (CPS $\Leftrightarrow$ R/C)
    - Frequency
  - CPS
    - Bus Signals
    - Timing
    - Structural Integrity

Effect Domain / Victim Element

- Cyber
  - CPS
    - Executable Code (Process)
    - Flash Memory
  - Remote Controller / Computer
    - Operating System
  - Maintenance Computer
    - Operating System
  - Environment
    - Other UAVs
    - WL Receivers
- Physical
  - CPS
    - Position
    - Orientation
    - Movement Direction
    - Movement Speed
    - Angular Velocities
    - Motor Thrust
    - Physical Component Life Time
    - Physical Component Structural Integrity
  - Environment (i.e., Ground, Buildings, Trees; Cars, Airplanes; People, Animals;)
    - Safety
    - Structural Integrity

Please note that the notion to distinguish between the influenced and the victim elements differ from the common cyber-security approach where both elements are generally considered to be identical. Based on the analysis of the identified attacks we can state that the influenced and the victim elements can (but not necessarily should) differ. Also we want to emphasize that an attack on a single influenced element can generally affect multiple victim elements. Further, please note that elements presented in both dimensions have partly different structure and elements. This reflects the asymmetry between the two dimensions.

Even though it is implicitly understood that cyber-attacks on CPS can lead to cyber or physical effects, to our knowledge nobody has formalized this. Therefore, we explicitly introduce the derivative of the proposed dimensions, which should describe the *Domain Propagation* of an attack (see enumeration below). This derivative contains all four possible transactions between cyber and physical domains acting as a target or an effect of the attack.

Domain Propagation (i.e., Target-to-Effect Domains)

- Physical-to-Cyber (P2C)
- Cyber-to-Cyber (C2C)
- Cyber-to-Physical (C2P)
- Physical-to-Physical (P2P)

We consider this derivative as extremely important for the analysis of chaining effects as well as for the attack characterization. Also please note that an attack can lead

to multiple domain propagation types at the same time. We see the cross-domain effects as the most cardinal distinction between attacks on CPS vs. attacks on computing or networking systems.

VI. Conclusion and Future Work

In this paper we have evaluated whether Data Flow Diagrams can be used for the systematic analysis of cyber-attacks on CPS. We have proved that it is possible but requires structural extensions of DFD. We have proposed extensions needed for the modeling and the subsequent analysis of the selected UAV. Nevertheless, we see that further investigations are needed in order to verify whether the proposed extensions are also sufficient for more complex cyber-physical systems.

One result of our analysis should be especially stressed. Even though we have focused on the cyber-attacks, applied to CPS they produce qualitatively different results. In about 1/3 of the identified attacks we have seen various "outbreaks" into the physical domain, including infliction of the physical damage to both CPS and its environment. Possible follow-up consequences like violations law/regulations or influence on the society increase the complexity even more.

As a consequence of our analysis we see that the following steps should be undertaken. First of all, the structure of Cyber-Physical Attacks on CPS should be analyzed and captured in a form of taxonomy. In Section V we have presented the preliminary results of our work going in this direction. We have identified that, considering attacks on CPS, it is necessary to distinguish between *Influenced* and *Victim Elements*. This distinction enables capturing and description of cross-domain effects, which are a distinct feature of attacks on CPS. After finalizing our work on the taxonomy dimensions, we plan to define a language for description of Cyber-Physical Attacks. Both taxonomy and attack description language should provide the basis for the elaboration of Cyber-Physical Attacks's knowledge base. We see such knowledge base as the necessary prerequisite for the better understanding of faced challenges.

The approach presented in this paper is suitable for the manual security assessment. The general drawbacks of the manual analysis are its strong dependence on the expert knowledge and the bad scalability. Therefore, we see the necessity to automatize the CPS vulnerability assessment process as much as possible. We consider transformation of xDFD into a form suitable for automatic analysis, e.g., petri nets. The existing petri nets simulation tools can provide reachability analysis of CPS states altered by various Cyber-Physical Attacks. The backtracking would provide the attack sequences leading to these states. This, in turn, would provide valuable insights necessary to make CPS ruggedized against Cyber-Physical Attacks.

Acknowledgement

This work is supported in part by the National Science Foundation (CNS-1035655, CCF-0820088), U.S. Army Research Office (AROW911NF-10-1-0005) and Lockheed Martin. The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies or endorsements, either expressed or implied, of the U.S. Government.

References

[1] M. Abrams and J. Weiss, "Malicious control system cyber security attack case study-maroochy water services," *Australia, NIST*, pp. 8–9, 2008.

[2] J. Slay and M. Miller, "Lessons learned from the maroochy water breach," *Critical Infrastructure Protection*, pp. 73–82, 2007.

[3] D. Albright, P. Brannan, C. Walrond, I. for Science, and I. Security, "Did stuxnet take out 1,000 centrifuges at the natanz enrichment plant?" Tech. Rep., 2010. [Online]. Available: http://isis-online.org/uploads/isis-reports/documents/stuxnet_FEP_22Dec2010.pdf

[4] N. Falliere, L. Murchu, and E. Chien, "W32. stuxnet dossier," *White paper, Symantec Corp., Security Response*, 2011.

[5] A. Cárdenas, S. Amin, and S. Sastry, "Research challenges for the security of control systems," in *Proceedings of the 3rd conference on Hot topics in security*. USENIX Association, 2008, p. 6.

[6] K. Koscher, A. Czeskis, F. Roesner, S. Patel, T. Kohno, S. Checkoway, D. McCoy, B. Kantor, D. Anderson, H. Shacham, and S. Savage, "Experimental security analysis of a modern automobile," in *Proc. IEEE Symp. Security and Privacy (SP)*, 2010, pp. 447–462.

[7] Y. Mo, T. Kim, K. Brancik, D. Dickinson, H. Lee, A. Perrig, and B. Sinopoli, "Cyber–physical security of a smart grid infrastructure," *Proceedings of the IEEE*, no. 99, pp. 1–15, 2012.

[8] J. Laprie, "Dependable computing and fault-tolerance," *Digest of Papers FTCS-15*, pp. 2–11, 1985. [Online]. Available: http://www.macedo.ufba.br/conceptsANDTermonology.pdf

[9] A. Avizienis, J.-C. Laprie, B. Randell, and C. Landwehr, "Basic concepts and taxonomy of dependable and secure computing," vol. 1, no. 1, pp. 11–33, 2004.

[10] S. Schneider, "Security properties and csp," in *Security and Privacy, 1996. Proceedings., 1996 IEEE Symposium on*. IEEE, 1996, pp. 174–187.

[11] D. M. Nicol, W. H. Sanders, and K. S. Trivedi, "Model-based evaluation: from dependability to security," vol. 1, no. 1, pp. 48–65, 2004.

[12] I. J. H. Marburger and E. F. Kvamme, "Leadership under challenge: Information technology r&d in a competitive world. an assessment of the federal networking and information technology r&d program," Tech. Rep., 2007.

[13] S. M. Rinaldi, J. P. Peerenboom, and T. K. Kelly, "Identifying, understanding, and analyzing critical infrastructure interdependencies," *IEEE Control Systems*, vol. 21, no. 6, pp. 11–25, 2001.

[14] S. M. Rinaldi, "Modeling and simulating critical infrastructures and their interdependencies," in *Proc. 37th Annual Hawaii Int System Sciences Conf*, 2004.

[15] E. Lee, "Cyber physical systems: Design challenges," in *Object Oriented Real-Time Distributed Computing (ISORC), 2008 11th IEEE International Symposium on*. IEEE, 2008, pp. 363–369.

[16] S. Burns, "Threat modeling: A process to ensure application security," *GIAC Security Essentials Certification (GSEC) Practical Assignment*, 2005. [Online]. Available: http://www.sans.org/reading_room/whitepapers/securecode/threat-modeling-process-ensure-application-security_1646

[17] S. Hernan, S. Lambert, T. Ostwald, and A. Shostack, "Threat modeling-uncover security design flaws using the stride approach," *MSDN Magazine-Louisville*, pp. 68–75, 2006. [Online]. Available: http://msdn.microsoft.com/en-us/magazine/cc163519.aspx

[18] S. Hansman and R. Hunt, "A taxonomy of network and computer attacks," *Computers & Security*, vol. 24, no. 1, pp. 31–43, 2005.

[19] C. Simmons, C. Ellis, S. Shiva, D. Dasgupta, and Q. Wu, "Avoidit: A cyber attack taxonomy," *University of Memphis, Technical Report CS-09-003*, 2009. [Online]. Available: http://issrl.cs.memphis.edu/files/papers/CyberAttackTaxonomy_IEEE_Mag.pdf

[20] Asctec hummingbird autopilot. [Online]. Available: http://www.asctec.de/asctec-hummingbird-autopilot-5/

[21] M. Howard, D. LeBlanc, S. T. B. Online, and S. B. O. (Firme), *Writing secure code*. Microsoft press Redmond, WA, 2003, vol. 2.

[22] A. Cárdenas, S. Amin, B. Sinopoli, A. Giani, A. Perrig, and S. Sastry, "Challenges for securing cyber physical systems," in *Workshop on Future Directions in Cyber-physical Systems Security. DHS*, 2009.