2024 IEEE International Automated Vehicle Validation Conference (IAVVC) | 979-8-3503-5407-2/24/$31.00 ©2024 IEEE | DOI: 10.1109/IAVVC63304.2024.10786397

A Generalized Stochastic Petri Net for Survivability and Security Analysis of Intelligent Transportation Systems

Justin L. King, Sahra Sedigh Sarvestani, and Ali R. Hurson
Missouri University of Science and Technology
Rolla, USA
Email: {jk3xf, sedighs, hurson}@mst.edu

***Abstract*—Cyber attacks continue to evolve in methodology and complexity, and pose a serious threat to safety and dependability of critical infrastructure, including transportation networks. This paper proposes a framework for modeling and simulating cyber attacks against vehicle-to-vehicle and vehicle-to-infrastructure communications. The framework facilitates observation and analysis of these attacks and enables assessment of security and survivability of autonomous vehicles reliant on this communication. A generalized stochastic Petri net is used to model IEEE 802.11p messaging between vehicles. The proposed model reflects the contention process for the enhanced distributed channel, which is often omitted in related models. Also reflected are attackers' actions to create a denial-of-service attack against a broadcast. The model is validated through respective simulations in GreatSPN and Vehicles In Network Simulation (Veins).**

***Index Terms*—survivability, cybersecurity, autonomous vehicles, Petri nets, intelligent transportation, Veins, IEEE 802.11**

I. Introduction

In the two decades since its adoption, Dedicated Short Range Communications (DSRC) has become the de facto standard for safety-related vehicle-to-vehicle (V2V) and vehicle-to-infrastructure (V2I) communications [1]. In the United States, safety messages are communicated through the Wireless Access in Vehicular Environments (WAVE) system at a frequency between 5.850-5.925 GHz [2], [3]. In this context, roadside units (RSUs) are the infrastructure component of V2I communication over WAVE/DSRC. IEEE 802.11p is the WAVE-related amendment to IEEE standard 802.11 for wireless local area networks, which was incorporated into the base standard in 2012 and updated in 2016 and 2019 [4]. In 2022, a new standard, 802.11bd was approved, incorporating key changes to improve 802.11p while maintaining backwards compatibility. These improvements are mainly focused on increasing transmission range for longer distances, increasing bandwidth needed for resource-heavy applications, and improving reliability. The approved use of the 60 GHz frequency band was one of the key components of 802.11bd, addressing the increasing need for higher bandwidth. Other modifications were 3 dB better receiver minimum input sensitivity and use of 20 MHz channels, rather than 10 MHz, as well as changes to the channel encoding scheme [4].

Funding from NSF DUE-1742523 and DUE-2221559 and DEd GAANN awards P200A180095 and P200A210121 is gratefully acknowledged.

Prior to the advent of this horizontal and vertical intelligent transportation communications infrastructure, hackers were already interested in attacking vehicles, mainly for name recognition. As connectivity and autonomy increase, cyber attacks on vehicles and the infrastructure are becoming more prevalent. One such attack is the recent Toyota hack, where customers' personal data was exposed for years before being discovered, impacting the in-vehicle device ID, map updates, and map data creation dates [5]. Such attacks can be used as a pivot point to impact additional components, subsystems, and more sophisticated system-of-systems attack vectors. Readers are referred to [6] for a detailed discussion of vulnerabilities, threats, and attacks associated with autonomous vehicles.

In this paper, we focus on analyzing the survivability of connected autonomous vehicles (AVs) against cyber attacks; specifically denial-of-service (DoS) attacks. We identified that DoS attacks were still a relevant threat vector, even to vehicular networks, over three years ago, when we began our research. As of last August, the Automotive Information Sharing and Analysis Center (Auto-ISAC) created an entry for DoS as an attack vector with a real-world example of the exploit of a Tesla Model-S [7]. Survivability is concerned with the transient behavior of a system from failure through recovery [8], [9]. We model and simulate an attacker conducting two varieties of DoS attacks against the network queue of the intelligent transportation system (ITS); we classify these as a naive DoS attack and a sophisticated DoS attack, respectively. The sophisticated attack includes additional spoofed messages to take advantage of priority access of the channel.

Figure I depicts the communication of safety messages in an intelligent transportation system. The dashed oval encompasses the lower tier communication that takes place amongst vehicles and between vehicles and the roadside, which is the focus of this paper. We model this communication as a generalized stochastic Petri net (GSPN). We chose to use a GSPN because of the modeling power of Petri nets (PNs), as well as the ability of GSPNs to model both exponentially distributed and deterministic transition firing rates.

The vehicles and infrastructure are simulated using Veins, configured as closely as possible to the real world, with the simulation environment based on the US Department of Transportation autonomous pilot initiatives [10], as well as

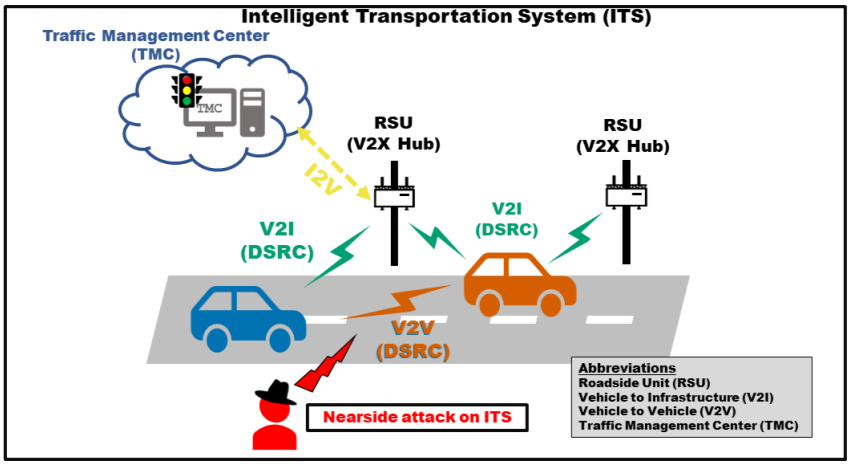

Fig. 1. Communication of Messages in an ITS

previous case studies where the same underlying software was used for vehicular testing [11][12]. Our simulation allows for real-time observation of effects of the attack on the ITS network infrastructure and vehicles, down to the protocol level where Enhanced Distributed Channel Access (EDCA) is leveraged during one of the attack scenarios that we tailored into our Veins configuration for that specific scenario.

To our knowledge, our research is the first study to model 802.11p/WAVE as a GSPN, the first to include analysis of EDCA towards improvement of security and survivability, and the first to model cyber attacks against 802.11p (Non-Next Generation Vehicle-to-Everything). In order to avoid confusion with the new standard, we continue to use the term 802.11p throughout the paper. Unlike related studies, e.g., [13], [14], [15], the simulation environment used for validation of our model is purpose-built to 802.11p specifications. Eckhoff et al. demonstrated that more general 802.11 simulators in high-density vehicle scenarios produce significant deviations when compared to an 802.11p simulator [16].

The major contribution of this paper is an analytical model that captures the behavior of the ITS network communications queue to include priority messaging with EDCA. We also define metrics to quantitatively assess survivability and security against cyber attack. The model is verified through respective simulations with GreatSPN and Veins. The latter reflects the operational environment. IEEE 802.11p EDCA message priorities and the cyber attacker's actions are reflected in both simulation environments.

II. Background and Related Works

Autonomous driving technology has been in development for over twenty years and is now being rapidly deployed. Throughout most of the twentieth century and early twenty-first century, cyber threats were not considered during vehicle design and manufacturing. This was a costly mistake, given that AVs are cyber-physical systems where sensors such as Global Positioning System (GPS) and radar, actuators, and embedded computers are typically interconnected with multiple communication paths. Their dynamic, episodically connected nature presents many challenges to researchers in the security and survivability domains. With this technology comes the great responsibility of ensuring these systems operate securely, provide trustworthy data, and are survivable in order to meet their mission requirements. The complexity of these systems, as well as the diversity of their communication paths, calls for new design and evaluation approaches to make vehicles more survivable against cyber attacks. Many well-publicized exploits of vehicle vulnerabilities have emphasized the critical role that cybersecurity has to ensure a more widespread adoption of AV technology through an operator's confidence in the safety and increased survivability of the vehicle [17][18].

Yin et al. model DSRC as a generalized M/G/1 queue, using a semi-Markov process (SMP) to analyze the medium contention window and backoff process [19]. Their follow-on work characterizes the basic safety message (BSM) in broadcast network operations, where they analyze the performance and reliability of the channel [20]. They ignored messages generated during the service channel interval, instead choosing to use a uniform distribution during the control channel interval only. Yao et al. [21] modeled 802.11p as an M/G/1/K queue with finite capacity, unlike preceding studies that assumed infinite capacity. However, they focused on control channel performability and reliability. Shah and Mustari modeled 802.11p using a one-dimensional Markov Chain [22]. Their study focused on performance analysis, includes EDCA with contention, and was validated through simulation in Matlab. They found similar intuitive findings to what we and others determined regarding the probability of packet transmission and throughput decreasing as the number of vehicles connected to the network increases.

Closer in relation to the modeling portion of our research, although there are vast differences, is the SPN proposed by Heindl and German for the 802.11 MAC protocol [23]. German and Heindl model 802.11 Medium Access Control protocol as a Stochastic Petri Net, specifically focusing on the distributed coordination function (DCF). The DCF is the

TABLE I
COMPARISON OF RELATED WORK TO THIS PAPER

Study	Protocol	Formalism	Subject of Model	Difference from This Paper
[19]	802.11p	SMP	CW and backoff	Did not account for queued requests
[22]	802.11p	1-D MC	EDCA process	Performance-based, Matlab verification
[21]	802.11p	2-D MC	EDCA M/G/1/K finite queue	Reliability- focused, validated with ns2 simulation
[21]	802.11p	1-D MC	Queuing system	Neglects impact of vehicle mobility on reception rate
[23]	802.11	SPN	Performance model of 802.11	Detailed, folded, decomposed SPN; see para for discussion

contention-based sublayer that performs the carrier sensing multiple access (CSMA) to prioritize frames, in order to avoid collisions. 802.11p is a broadcast protocol, which uses the Enhanced Distributed Channel Access (EDCA) mechanism (developed as a replacement to DCF) with different Arbitration Inter Frame Space (AIFS) and Contention Window (CW) values used to set user priorities. These categories are prioritized in our SPN model, and were not defined until the IEEE 802.11-2012 release, more than eleven years after the Heindl and German paper was published. Their work also did not include timing synchronization and they assumed ideal channel conditions. We are unaware of any other research that models 802.11p as a SPN with timing synchronization, including EDCA, without ideal channel conditions. That is an important part of this research to begin analysis, but not the sole focus. Table I summarizes the aforementioned studies and compares them to our proposed approach.

III. METHODOLOGY

AVs have many different interacting and interdependent components, which enable the propagation of an attack, up to and including failure, once an attacker gains a foothold. The overarching goal of our research is to analyze survivability and security of AV and its operating environment in the face of an active cyber attack. We model layers 1 and 2; Physical (PHY) and Medium Access Control (MAC) layers of the IEEE WAVE Architecture as a GSPN, where tokens represent messages which nodes (vehicles or roadside units) broadcast throughout the ITS, enabling V2V and V2I communications. Our GSPN models the foundational level communications for the ITS, which we logically designed matching the same protocol stack that is coded into VEINS simulation, allowing for verification of the two models. In this paper, we consider an adversary that is flooding a multi-vehicle broadcast domain with packets to create a denial-of-service. We portray two varieties of DoS attacks against two different scenarios, comprising four different attack vectors. The attacks are simulated using our GSPN model in one software environment (GreatSPN) compared to a high fidelity vehicular testing software (Veins). Once simulation is complete, we evaluate the results both quantitatively and qualitatively. This initial paper focuses on 802.11p, given that it still plays a major role in AV communications. We will add to the network attack surface and threat vectors in follow-on research, and hope to include 802.11bd.

In the six decades since their inception, Petri nets have facilitated both functional and non-functional analysis of complex systems. At the basic level, a Petri net consists of places, transitions, arcs, and markings (tokens). Transitions, which represent events, are depicted as bars. Places represent conditions and are depicted as circles. Each place may contain a discrete number of marks (tokens). The default capacity of a place is infinite. The marking of a Petri net at a given time is the assignment of tokens to its place. An arc connects a place to transitions. A transition "fires" if it is enabled, i.e., if it has at least the required number of tokens in each of its input places. The required number is the weight of the arc. Inhibitor arcs designate transitions that are enabled by the absence (not presence) of a specific number of tokens. Firing, which is an atomic step, transfers the required number of tokens from one or more input places to one or more output places of the transition according to arc weights. In the absence of an execution policy, if multiple transitions in a Petri net are enabled, the order in which they fire is non-deterministic.

An enhancement proposed by Chiola et al. enables the definition of different priority levels for immediate transitions [24]. They define a Petri net as the seven-tuple in Equation 1.

$$GSPN = (P, T, \Pi(.), W^{-}(.), W^{+}(.), W^{H}(.), \Lambda(.), M_0) \tag{1}$$

In this equation, places and transitions are denoted as P and T, respectively. The priority function $\Pi(.)$ assigns priorities to transitions, with 0 being the default for immediate transitions and 1 the default for timed transitions. Input, output, and inhibitor arcs are $W^{-}(.)$, $W^{+}(.)$, and $W^{H}(.)$, respectively. $\Lambda(.)$ allows for the stochastic component of the model, defining the rate of transition. M_0 is the initial marking of the Petri net.

The following assumptions underlie our model:

- We assume an alternating WAVE access scheme for all vehicular representations in our simulations, where one radio transceiver is used for both transmitting and receiving messages. In this scheme, defined in [25], the radio alternates between the control channel (CCH) and service channel (SCH) every 50 ms, with a guard interval of 4 ms. Beacons are sent on the control channel every 50ms, while application messages are allowed on service channels. This method uses the entire 100ms timeslot.
- We assume that all vehicles in the PN have the same transmit and receive ranges within the broadcast domain.
- Transmission power is not taken into account, neither is channel shadowing or fading, or vehicle mobility.

The current model is subject to the following limitations:

- We are currently limited to portraying attacks in simulation, meaning we are not able to conduct these ex-

periments on an actual vehicle. This is quite often the case for vehicular cybersecurity research; given the high monetary cost to test, the risk of bodily harm or death to humans when testing, and the potential vehicle damage and/or destruction due to cyber threat portrayal. We plan to extend our work in the near future to include sensor data feeds, which would allow for additional research into cyber attacks aimed at impacting autonomous features.

- Contention window values were implemented using the minimum value specified, rather than a range of values due to software limitations. We tested both minimum and maximum CW values and achieved better results with maximum CW values, since a larger CW value creates longer interframe spacing. Therefore, we chose to use the CW minimum values to err on the side of worst case scenario, where availability was lower.

A. Petri Net Model

In their seminal work on modeling power hierarchy, [26] consider state space models to be more powerful than non-state space, especially when capturing dependencies. GSPNs are at the top of this power hierarchy, and are analogous to Continuous Time Markov Chains (CTMCs). GSPNs can be converted to obtain the underlying CTMC and vice versa. A GSPN has both negative exponentially distributed firing weights and deterministically zero firing rates. Figure 2 captures both our base 802.11p/WAVE communications model, and adds the attacker place to the scenario, one figure is shown, given space limitations. The nomenclature for the places and transitions, respectively, are presented in Tables II and III, respectively.

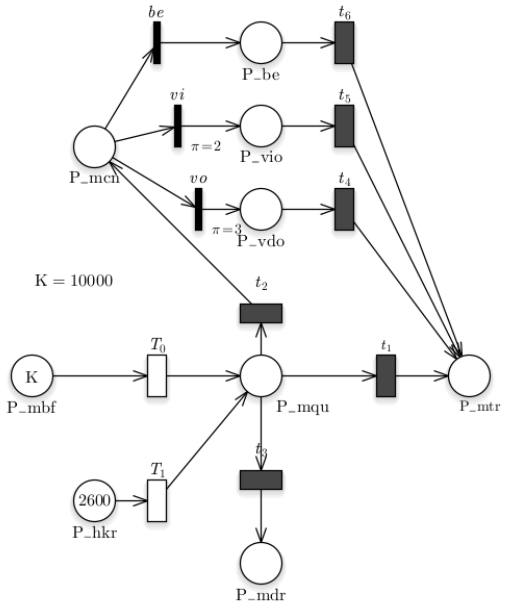

Fig. 2. Petri Net Model of 802.11/WAVE with Attacker

TABLE II
PETRI NET PLACES

Place	Description
P_{mbf}	Message Buffer
P_{mqu}	Message Queue
P_{mcn}	Message Contention
P_{be}	Best Effort
P_{vio}	Video
P_{vdo}	Voice (Data) Only
P_{mtr}	Transmit Message
P_{hkr}	Attacker Message Queue
P_{mdr}	Drop Message Queue

TABLE III
PETRI NET TRANSITIONS

Transition	Description
T_0	Queue message
T_1	Hacker fires message
t_2	Message Contention
t_3	Channel Unavailable
t_{be}	Message is best-effort priority
t_{vi}	Message is video priority
t_{vo}	Message is voice-only priority
$t_{4,5,6}$	Contention resolved, fire to transmit
t_1	Message fires to transmit

Both the arrival rate and service rate distributions are exponentially distributed; messages arrive into the buffer at a rate of λ, and the initial service firing rate μ. The attacker's firing rate is exponentially distributed. The number of tokens representing the messages are listed on each of the PN figures. All other transitions are either immediate or deterministic; a thin bar represents an immediate transition firing, occurring in zero time, while a thicker bar represents a deterministic firing rate, with each transition being timed input based on 802.11p timing parameters. We verify our GSPN model through a combination of quantitative analysis directly from the PN model and simulative analysis compared against VEINS results.

GreatSPN is a suite of tools used to model GSPNs and Stochastic Well-formed Nets (SWNs), it has been in development since the 1980s with steady improvements to enhance the GUI and analytical capability [27]. We chose GreatSPN because it offers the ability to specify priorities on transitions, where other available SPN simulation software did not give as much granular control. This specific component of the software enabled the ability to simulate EDCA user priorities that we designed into our model.

B. Petri Net Metrics

When dealing with complex systems, the reachability graph suffers from state explosion, creating the need to use simulation for assessing the model [28]. The other contributing factor is due to the large number of tokens being generated in both normal traffic and attacker scenarios, resulting in an enormous reachability graph (RG). The RG was used in our quantitative analysis; we also analyzed the PN structurally including P and T semi-flow relationship invariance.

Equation 2 shows the probability that T_i wins the race and fires, given marking M_k. $W(T_i)$ is the weight of a tangible marked transition in this case, however it can be used to represent vanishing transitions as well with $W(t_i)$. While $E(M_k)$ represents the set of enabled transitions in marking M_k.

$$P\{T_i|M_k\} = \frac{W(T_i)}{q_i} \text{where} q_i = \sum_{T_{(i,j)} \in E(M_k)} W_{(i,j)} \quad (2)$$

The communication system is considered available if transitions $t_1, t_4, t_5,$ or t_6 fire.

Degraded operations occur when transition t_2 fires, creating a contention scenario: $D_{ops} = t_{2out}$, where t_{2out} is the multiplicity of the output arc from transition t_2 to P_{mcn}. This is when AIFS message priorities impact the outcome of a message exiting the queue. An attacker's success (PrA_{Su}) is dependent upon the number of tokens in place P_{hkr} divided by the sum of messages that arrive at the buffer (P_{mbf}), while removing the messages that drop (P_{mdr}), either due to contention or because of the impact of the attacker on the network.

$$P\{A_{su}\} = \frac{M_{P_{hkr}}}{M_{P_{mbf}} - M_{P_{mdr}}} \quad (3)$$

C. AV Simulation Environment

The simulation environment consists of the Vehicles in Network Simulation (Veins) framework and related software such as the Objective Modular Network Testbed in C++ (OMNeT++) and Simulation of Urban MObility (SUMO). Veins is a vehicular network simulation framework based on OMNeT++ and SUMO, which are a network simulation environment and microscopic traffic simulation package respectively [29]. Veins simulations are carried out by running NED files, which are native to OMNeT++ and describe simulation scenarios. Steps for running a Veins attack scenario can be found in [30].

An "Instant Veins" solution is offered by Veins developers as a virtual machine with Veins pre-installed and ready to use OOTB. Veins can also be manually installed and setup by users on their own machines. In this research, we used both Instant Veins and a Linux machine with Ubuntu 22.04.01 to run Veins. The Instant Veins version used is Instant Veins 5.2-i1 which contains Veins 5.2, INET Framework 4.2.8, OMNeT++ 5.7, and SUMO 1.11.0. The OS is Debian Linux 11; the Linux machine in our own lab setup consists of Veins 5.2, SUMO 1.8.0, and OMNeT++ 6.0. There were no differences noted in simulation results based on OS or software version differences.

The parameters in Table IV are used in the V2V attack scenarios (naïve and priority) without an RSU. The last two parameters in Table IV are used in the V2I attack scenarios (naïve and priority) with an RSU to facilitate ITS attack scenarios. Veins is suited for 802.11p testing, given the inclusion of a MAC/PHY 802.11p model specifically intended for vehicular networks. Veins has the ability to perform channel switching and model Enhanced Distributed Channel Access (EDCA) queueing. Packets are assigned EDCA priorities via their assigned Access Category (AC). Higher ACs gain higher priority in channels. ACs include:

- AC_BK (Background)
- AC_BE (Best Effort)
- AC_VI (Video)
- AC_VO (Voice Only)

Where AC_BK is the lowest priority and AC_VO is the highest. Veins uses an EDCA Function (EDCAF) which controls queue back-off and transmission initiation [31].

TABLE IV
V2V-V2I SCENARIO PARAMETERS

Parameter	Value
No. of Vehicles	100
No. of Attackers	1
Total Simulation Time	200s
Attack Time Start	57s
Attack Duration	26s
Attack Beacon Interval	1s
Transmission Power	20mW
No. of RSUs	1 (V2I)
RSU Beacon Interval	1s (V2I)

D. Veins Expanded Tests-Attacks Description

We chose two types of Denial of Service (DoS) attacks to simulate. The first is a "naïve" attack (NV2V in Table VI), where the attacker floods the network with packets. The second is a "priority" attack (PV2V in Table VI), where the attacker floods the network with packets that are set to a higher priority. Veins does not simulate attack scenarios out of the box but offers functionality to achieve this purpose. Furthermore, each attack scenario is run with and without an RSU in the simulation. Thus, a total of four attack scenarios are simulated in Veins. The example scenarios and nodes (e.g., Car node) native to Veins were used to simulate the traffic flow and "normal" (non-attack) traffic. The purpose of this research is to observe how an attack disrupts normal operation, so it is not necessary to deviate from the given method of normal traffic generation/simulation. The attacker is built off of Veins' native Car node and is modelled in simulation as a vehicle. SUMO contains the Traffic Control Interface (TraCI) module, which allows users to access and modify the traffic simulation [32]. Nodes in Veins use TraCI to modify application layer behavior. The TraCI module for the attacker is defined to flood the network with packets during the entirety of the attack. As stated, in the naïve attack, the attacker floods the network without further configuration needed. In the priority attack, the priority of the attacker's messages are configured to be higher priority in the simulation's .ini file. Veins natively configures messages to the highest AC, AC_VO, so for this attack normal traffic was reconfigured to be the lowest access category, AC_BK, and attack traffic was configured to the highest access category .ini files are an OMNeT++ concept which allow users to configure simulation parameters such as total simulation time, application types for nodes, message frequency and priority, and so forth [33].

The attacker is spawned and despawned into a simulation via a SUMO configuration file that defines traffic flow and management.

E. VEINS Metrics

Result collection is done natively by Veins and OMNeT++ and results are available as .vec, .vci, and .sca files, corresponding to a vector file, vector index file, and scalar file respectively. Results can be viewed via the OMNeT++ IDE, but users can also export results as different file types (e.g., .csv). There are many results that Veins is automatically programmed to collect, some of interest include total packet loss, total messages/packets sent, start/stop time of nodes (e.g., Car and Attacker nodes), messages sent/received by each node, and more. The following metrics are of interest to this research:

- c_0 Time that DoS occurred; we define as > 3 consecutive broadcasts from attacker
- c_p Time to restore partial communications
- c_R Time to restore all nodes; all communications restored
- b_c Total messages broadcast, exponentially distributed with rate λ
- b_r Total messages received, only known at RSU
- b_d Total messages lost/dropped; deterministic firing rate of transition t_2

There are two main nodes of interest in the simulation, the attacker node and the car node (labeled as "node" and identified with a number) in Figure 3, as well as the RSU node for scenarios that involve an RSU. There are results from four different layers/modules of interest for attacker and car nodes: application layer, NIC Phy80211p, NIC MAC 1609.4, and the VeinsMobility module. Results from specific nodes and specific modules are selected to compute the above metrics. The mapping of metrics to Veins results listed in Table V.

Veins does not collect data on node failures and node restorations. We added result collection to determine specific times in the simulation when packets were dropped by a given node. A vector of drop times was recorded and available in the results after a scenario was run. Only assuming a packet dropping is a failure would create too many up and down scenarios. We used the average time that the channel was busy to indicate the mean-time-to-repair (MTTR) in the attacker scenarios. Total messages broadcasted, received, and lost are metrics which are already collected by Veins and are described in further details in the next section.

TABLE V
MAPPING OF METRICS TO VEINS RESULTS

Metric	Veins Result	Node	Module
c_0	DoS	attacker	N/A
c_p	3.70 s	all	VEINSwAtt
b_c	SentPkts	node	Appl
b_r	RcvdBcasts	node/RSU	nic.mac1609_4
b_d	LostPkts	node	nic.mac1609_4

IV. SIMULATION RESULTS ANALYSIS

A. Petri Net Results

The attack scenarios were previously described in Section III-D; these are the results for the Petri net (GSPN) under normal network conditions, using a timed PN with random automatic firing, which automatically runs the simulation without constant human input. The probability of successful packet transmission, p_{tx}, was found to be 93.16%, leaving 6.84% as the probability of unsuccessful communication. Once the attacker is introduced, using the scenario described earlier in this paper, the probability of unsuccessful communication (failure rate) increases to 12%, nearly twice the pre-attack rate, and over the 10% mark defined by IEEE 1609 as the maximum failure rate allowed.

Bai and Krishnan [34] developed the τ window availability metric to capture the reliability of vehicular applications, given their wide variation in inter-arrival times. The idea behind this was that as long as one packet from a neighbor vehicle is successfully received within a tolerance time window, the receiver vehicle should be able to predict and update the neighbor vehicle's information accurate enough for vehicular application processing. Connected vehicles' application tolerance windows varies from 3.0 ms-3.0 s, depending on the type of application. These include such critical applications as forward collision warning (FCW) and stopped vehicle ahead (SVA). For this analysis, we selected the minimum τ window of 3.0 ms, for the worst case scenario. Given broadcast communications, packets arrive with rate λ, independent of other packets (time homogeneity). The splitting rate is $\lambda\rho$.

$$P\{Su_{tx}\} = \rho\lambda\tau = .88(.10).003 = .026 \tag{4}$$

where ρ =Probability Successful Transmission
and λ =Packet Arrival Rate

During the 300ms τ window, using our attacker Petri net data analysis, a legitimate vehicle only has a .03% probability of message survival (transmission).

TABLE VI
AVAILABILITY ANALYSIS RESULTS

Parameter	Availability	Unavailability
PN Base	86.61%	13.39%
PN w Attacker	90.37%	9.63%
VEINS NV2V	98%	2.0%
NV2V RSU	97.8%	2.2%
RSU	90.3%	9.7%
PV2V	98.7%	1.3%
PV2V RSU	98.7%	1.3%
RSU	93.2%	6.8%

B. Veins Results

The data captured from each of the Veins tests is displayed in Table VII, beginning with a baseline run of the simulation for reference, then each of the four test scenarios described in the previous section. The Erlangen route was used, which is

TABLE VII
VEINS DATA AND RESULTS

Test	Total Sent	Total Generated	Total RcvDd Bcast	Total Msgs Rcvd	Total Msgs Lost	Failure Rate
Baseline	4,243	4,243	42,110	42,110	5,039	11.97%
FullNV2V	83,874	20,173	1,081,518	1,081,518	48,712	4.52%
FullNV2VRSU	105,327	48,183	1,040,175	1,040,175	2,292,193	28.71%
Full PV2V	56,674	49,544	738,889	104,763	33,399	4.52%
Full PV2V RSU	67,716	171,214	718,665	115,987	149,539	20.34%

notable because line of sight challenges created by buildings on this route lead to propagation errors.

A visual of the simulation, in Figure 3, shows one RSU and thirteen nodes; nodes enter the simulation just above node[13].

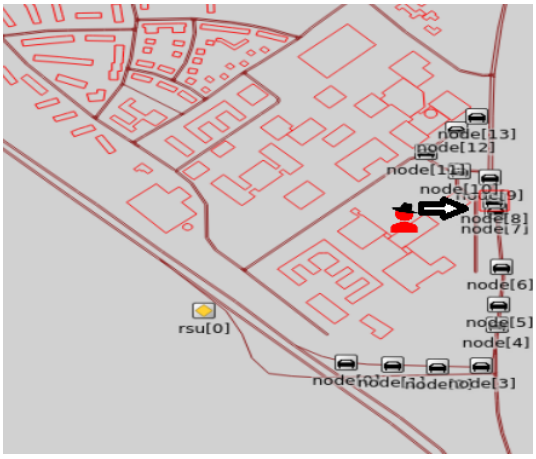

Fig. 3. Veins Simulation

C. Validation

Comparing the Petri net simulation results to the results that we saw in our Veins simulation, the analysis shows that our PN predicted the probability of transmission within just over 5% of the baseline of Veins. The main reason for the PN's higher transmission probability prediction is because the signal to noise plus interference (SINR) ratio not being factored into the model. This is something that we would like to add in the future. We predicted availability within 99.9% accuracy when comparing attack results from the PN against VEINS simulation with the RSU, and 97% accuracy with priority messaging against the RSU. These settings could be tweaked within the PN to change the contention window to obtain 99.9% accuracy. The reason comparing to the RSU attack is significant is because a DoS on the RSU would take out a significant portion of the ITS, leaving individual vehicles to attempt communication without a central location to parse messages. Regarding contention window, the longer the CW size, the greater the probability of transmission; availability decreases, so there is a delicate balance. The other factor that cannot be replicated exactly in the model is the density of vehicles in the simulation, the Veins simulation has sixty-seven vehicles at very close distances, competing for time slots. We attempted to replicate 802.11p and the simulation environment in the PN model as closely as possible, even firing 10,000 tokens through the PN. Shah and Mustari found in their research as well that as the number of vehicles increases, the transmission probability and throughput for each access channel queue reduces [22]. Li et al. similarly observed that the application level reliability meets quality of service requirements for all vehicles within 100m range when vehicle density is less than 0.35veh/m however, as the density increases, a longer tolerance time window is needed to meet QoS reliability requirements [35].

V. CONCLUSION

In this paper, we described our analytical model, which captures the behavior of vehicular communications to include priority messaging with EDCA. This model was simulated in GreatSPN and key metrics were derived to quantitatively measure survivability and security. The results of the event simulation were compared against the analytical model to evaluate the survivability of vehicular communications over WAVE. Current research stops short of analyzing and measuring the impact that specific vulnerabilities in network design have on survivability when an attack is executed. Our adversarial portrayal has been very deliberate, beginning with research and classification of attacks in an attack database, stored in an attack defense tree. The same holds true for other attributes. Previous research did not include survivability, multiple channels, or security. To our knowledge, research does not exist where 802.11p characteristics are modelled as a GSPN. Our work is the first to model the 802.11p enhanced distributed control access process, focusing on the security and survivability aspects of the protocol and V2X communications.

Our proposed model has broad applicability and can be adapted to any variant of 802.11x protocol. The Petri net models an attacker's actions against the broadcast medium, which is useful in many different scenarios outside the automotive domain. Another goal is to advance further ahead of the developmental lifecycle by identifying where an attacker might succeed early, enabling manufacturers to identify and fix vulnerabilities, thereby increasing survivability. Research in cybersecurity and survivability is needed to ensure AVs meet their mission, our work analyzes the impact of attacks on the system and strives to increase these attributes through our analysis and recommendations. We plan to add multiple receivers in the future to simulate vehicles and the impact of SINR on the channel's survivability and security.

This work would be useful from a scientific perspective because it can be repeated and adapted as innovative technology is added onto autonomous vehicles, to include electric vehicles. This allows early developmental process analysis and more informed design changes, which will increase AV survivability. Our research identifies weaknesses in AVs, which will be beneficial to both researchers and practitioners by informing defense in depth, and more secure vehicular design choices. Our models and simulation environment could easily be adapted to work with any variant of 802.11x protocol or for wired media as well.

We learned that a single attacker with a small number of tokens in the PN model can drop the packet error rate below the 90% threshold. Another interesting insight was the impact of the amount of vehicles and distance between vehicles, which negatively affected communications in Veins. Intuitively, implementing priority communications improved packet transmission rates, even with the attacker on the network, however the attacker created a DoS on the most critical asset, the RSU for over 19 s. In future work, we would like to address the density of vehicles by adding queues, then analyze the impact on communications once the number of nodes significantly increases. We are also researching options for a path forward to implement other types of WAVE messages, giving us the ability to simulate man-in-the-middle attacks.

REFERENCES

[1] Y. L. Morgan, "Notes on DSRC & WAVE standards suite: Its architecture, design, and characteristics," *IEEE Communications Surveys & Tutorials*, vol. 12, no. 4, pp. 504–518, 2010.

[2] NHTSA, "Federal motor vehicle safety standards; V2V communications," Jan. 2017. [Online]. Available: https://www.govinfo.gov/content/pkg/FR-2017-01-12/pdf/2016-31059.pdf

[3] Z. Liu, T. Liang, J. Guo, and L. Zhang, "Priority-based access for DSRC and 802.11p vehicular safety communication," in *Proc Intl Conf on Connected Vehicles and Expo (ICCVE)*, 2012, pp. 103–107.

[4] "IEEE Standard for Information Technology–Telecommunications and Information Exchange between Systems Local and Metropolitan Area Networks–Specific Requirements Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications Amendment 5: Enhancements for Next Generation V2X," 2022.

[5] I. Arghire. (2023, Jun.) Toyota discloses new data breach involving vehicle, customer information. Security Week. [Online]. Available: https://www.securityweek.com/toyota-discloses-new-data-breach-involving-vehicle-customer-information/

[6] J. L. King, E. Jackson, C. Brinker, and S. Sedigh Sarvestani, "Wheel tracks, rutting a new oregon trail: A survey of autonomous vehicle cybersecurity and survivability analysis research," in *Advances in Computers*, ser. Advances in Computers, A. R. Hurson, Ed. Elsevier, 2023, vol. 130, pp. 67–106.

[7] Auto-ISAC, "Automotive Threat Matrix-DoS," Aug. 2023. [Online]. Available: https://atm.automotiveisac.com/technique/Denial%20of%20service

[8] K. S. Trivedi and A. Bobbio, *Reliability and Availability Engineering: Modeling, Analysis, and Applications*. Cambridge University Press, 2017.

[9] M. Woodard, K. Marashi, S. Sedigh Sarvestani, and A. R. Hurson, "Survivability evaluation and importance analysis for cyber–physical smart grids," *Reliability Engineering & System Safety*, vol. 210, p. 107479, June 2021.

[10] U.S. Department of Transportation Federal Highway Administration. (2023) Intelligent transportation systems connected vehicle pilots. [Online]. Available: https://www.its.dot.gov/pilots/index.htm

[11] L. Ming, G. Zhao, M. Huang, X. Kuang, J. Zhang, H. Cao, and F. Xu, "A general testing framework based on Veins for securing VANET applications," in *Proc IEEE SmartWorld*, Oct. 2018, pp. 2068–2073.

[12] F. Ahmad, A. Adnane, V. Franqueira, F. Kurugollu, and L. Liu, "Man-in-the-middle attacks in vehicular ad-hoc networks: Evaluating the impact of attackers' strategies," *Sensors*, vol. 18, no. 11, p. 4040, Nov. 2018.

[13] M. Wellens, B. Westphal, and P. Mahonen, "Performance evaluation of IEEE 802.11-based WLANs in vehicular scenarios," in *Proc IEEE Vehicular Technology Conf (VTC Spring)*, 2007, pp. 1167 – 1171.

[14] T. Hecker, J. Zech, B. Schäufele, R. Gräfe, and I. Radusch, "Model car testbed for development of V2X applications," *Journal of Communications*, vol. 6, p. 2011, 02 2011.

[15] B. Liu, B. Khorashadi, D. Ghosal, C.-N. Chuah, and M. H. Zhang, "Assessing the VANET's local information storage capability under different traffic mobility," in *Proc IEEE INFOCOM*, 2010, pp. 1–5.

[16] D. Eckhoff, C. Sommer, and F. Dressler, "On the necessity of accurate IEEE 802.11p models for IVC protocol simulation," in *Proc IEEE Vehicular Technology Conf (VTC Spring)*, may 2012, pp. 1–5.

[17] C. Valasek and C. Miller. (2014) Adventures in automotive networks and control units. IOACTIVE. [Online]. Available: https://ioactive.com/resources/library/

[18] Z. Cai, A. Wang, W. Zhang, M. Gruffke, and H. Schweppe, "0-days & mitigations: roadways to exploit and secure connected BMW cars," *Black Hat USA*, vol. 2019, no. 39, p. 6, 2019.

[19] X. Yin, X. Ma, and K. S. Trivedi, "An interacting stochastic models approach for the performance evaluation of DSRC vehicular safety communication," *IEEE Trans Computers*, vol. 62, no. 5, pp. 873–885, May 2013.

[20] X. Yin, X. Ma, K. Trivedi, and A. Vinel, "Performance and reliability evaluation of BSM broadcasting in DSRC with multi-channel schemes," *IEEE Trans Computers*, 08 2013.

[21] Y. Yao, L. Rao, and X. Liu, "Performance and reliability analysis of IEEE 802.11p safety communication in a highway environment," *IEEE Trans Vehicular Technology*, vol. 62, no. 9, pp. 4198–4212, Nov. 2013.

[22] A. F. M. Shahen Shah and N. Mustari, "Modeling and performance analysis of the IEEE 802.11p Enhanced Distributed Channel Access function for vehicular network," in *Proc IEEE Future Technologies Conf (FTC)*, Dec. 2016, pp. 173–178.

[23] A. Heindl and R. German, "Performance modeling of IEEE 802.11 wireless LANs with stochastic Petri nets," *Performance Evaluation*, vol. 44, no. 1, pp. 139–164, 2001, performance and Dependability Techniques and Tools.

[24] G. Chiola, M. Marsan, G. Balbo, and G. Conte, "Generalized stochastic Petri nets: a definition at the net level and its implications," *IEEE Trans Software Engineering*, vol. 19, no. 2, pp. 89–107, Feb. 1993.

[25] "IEEE standard for wireless access in vehicular environments (WAVE) – multi-channel operation, IEEE Std 1609.4-2016 (revision of IEEE Std 1609.4-2010)," 2016.

[26] M. Malhotra and K. Trivedi, "Power-hierarchy of dependability-model types," *IEEE Trans Reliability*, vol. 43, no. 3, pp. 493–502, Sep. 1994.

[27] S. Baarir, M. Beccuti, D. Cerotti, M. De Pierro, S. Donatelli, and G. Franceschinis, "The GreatSPN tool: recent enhancements," *ACM SIGMETRICS Performance Evaluation Review*, vol. 36, no. 4, pp. 4–9, Mar. 2009.

[28] M. Marsan, G. Balbo, G. Conte, S. Donatelli, and G. Franceschinis, *Modelling with Generalized Stochastic Petri Nets*, ser. Series in Parallel Computing. West Sussex, England: John Wiley & Sons Ltd., 1995.

[29] C. Sommer, R. German, and F. Dressler, "Bidirectionally Coupled Network and Road Traffic Simulation for Improved IVC Analysis," *IEEE Trans Mobile Computing (TMC)*, vol. 10, no. 1, pp. 3–15, January 2011.

[30] Sommer, Christoph, "Tutorial - Veins," https://veins.car2x.org/tutorial/, 2021, accessed: 2022.

[31] D. Eckhoff and C. Sommer, "A multi-channel IEEE 1609.4 and 802.11p EDCA model for the Veins framework," in *Proc 5th ACM/ICST Intl Workshop on OMNeT++ (OMNeT++ 2012)*, Mar. 2012.

[32] German Aerospace Center (DLR). (2023) TraCI Sumo documentation. [Online]. Available: https://sumo.dlr.de/docs/TraCI.html

[33] OpenSim Ltd. (2019) A quick overview of the OMNeT++ IDE. [Online]. Available: https://omnetpp.org/documentation/ide-overview/

[34] F. Bai and H. Krishnan, "Reliability analysis of DSRC wireless communication for vehicle safety applications," in *Proc IEEE Intelligent Transportation Systems Conf (ITSC)*. IEEE, 2006, pp. 355–362.

[35] W. Li, J. Wu, X. Ma, and Z. Zhang, "On reliability requirement for BSM broadcast for safety applications in DSRC system," in *Proc IEEE Intelligent Vehicles Symp*, Jun. 2014, pp. 946–950.