VANETsim: An open source simulator for security and privacy concepts in VANETs

Andreas Tomandl, Dominik Herrmann,
Karl-Peter Fuchs and Hannes Federrath
University of Hamburg
Germany

Florian Scheuer
University of Regensburg
Germany

***Abstract*—Aside from massive advantages in safety and convenience on the road, Vehicular Ad Hoc Networks (VANETs) introduce security risks to the users. Proposals of new security concepts to counter these risks are challenging to verify because of missing real world implementations of VANETs. To fill this gap, we introduce VANETsim, an event-driven simulation platform, specifically designed to investigate application-level privacy and security implications in vehicular communications. VANETsim focuses on realistic vehicular movement on real road networks and communication between the moving nodes. A powerful graphical user interface and an experimentation environment supports the user when setting up or carrying out experiments.**

***Keywords*—VANET, Car2Car, Vehicular Communication, Simulator, Security, Privacy, Anonymity, Intrusion and Attack Detection, Privacy-Enhancing Technology, Security in Mobile and Wireless Networks**

I. INTRODUCTION

Vehicular Ad Hoc Networks (VANETs) have the potential to dramatically increase the safety on the road network by sharing telematic data like position, speed and direction of vehicles (so-called *beacons*) and warnings about dangerous situations such as crashes, traffic jams, and aquaplaning [1]. However, VANETs will also introduce new security issues. Several attacks on privacy and security have been identified in the scientific literature and a number of countermeasures have been proposed, e. g. [2]–[9] and [10].

Even though car manufacturers and research institutes already work with first prototypes to test and improve communication links and early driver assistant systems, no widely available testbeds or specifications for communications on higher levels exist at the moment. Therefore, researchers that want to evaluate the effectiveness of their proposals to improve security and privacy have to rely on abstract models and simulations of road and data traffic.

We introduce VANETsim (http://vanet-simulator.org), a vehicular communications simulator especially tailored for analyzing security and privacy concepts on the application layer, while integrating realistic vehicular movement and abstract network communication in one tool.

Contributions. Over the course of the last five years we have developed VANETsim, an open source simulation platform for security and privacy concepts in VANETs. VANETsim is an intuitive tool that allows researchers to evaluate their proposals and compare them with related work efficiently. In this paper we provide a concise description of its main features and announce its availability to the community. VANETsim provides the following distinctive features that distinguish it from other simulation approaches:

- To the best of our knowledge, VANETsim is the first simulator especially designed to analyze attacks on security and privacy as well as countermeasures on the *application layer*. New attacks and countermeasures can be integrated easily because the API provides high-level access to all relevant data structures.
- The included *Scenario Creator* allows to design repeatable experiments with varying parameters and to run batch experiments with one click.
- VANETsim provides a close approximation of the real world in order to obtain realistic results. It employs a micro-traffic model that simulates driving decisions of individual cars and allows to import road networks from *OpenStreetMap* (http://www.openstreetmap.org).
- The simulator offers a high level of *control*: besides realistic simulations, artificial scenarios (e. g., with simple graphs as road networks) can be created to verify analytic results.
- VANETsim allows for large-scale experiments in real time with more than 16,000 vehicles and large road networks on off-the-shelf desktop hardware. Its code is highly optimized and supports multi-core systems.
- In addition to a lightweight terminal interface we have also designed a powerful GUI that visualizes the simulation, and thus allows for explorative experimentation with intuitive debugging facilities.

This paper is structured as follows: Section II reviews existing simulation approaches for VANETs while Sect. III offers a high-level overview of our simulator. The individual components are described in Sect. IV. We proceed by discussing the *Scenario Creator* and the *Post Processing Engine* in Sect. V and the GUI in Sect. VI. Finally, in Sect. VII we present results from a performance evaluation of VANETsim with four large-scale experiments before we conclude the paper in Sect. VIII.

II. RELATED WORK

There are several universal simulation tools for vehicular networks available like TraNS [11], Veins [12]–[14] and VanetMobiSim [15]–[17] which provide interfaces to connect the road traffic simulators SUMO [18] or CanuMobiSim (http://canu.informatik.uni-stuttgart.de/mobisim/) to common computer network simulators like ns-2 (http://www.isi.edu/nsnam/ns/) or OMNeT++ [19], [20]. These approaches focus on widely used simulation tools that are pretty mature, target the network layer and provide a multitude of settings, but lead to a fairly complex configuration. Other simulators like GrooveNET (http://mlab.seas.upenn.edu/groovenet/, formerly called GrooveSim [21]–[23]) integrate different traffic and network models into one tool, but focus on the network layer only and concentrate on the supply of less but more specific features (e. g., hybrid simulations with real and virtual vehicles in GrooveNet). This leads to a more simple configuration, but limits their use to specific groups of researchers.

We believe that future research in the area of VANETs will benefit from a simulation engine that focuses specifically on the distinctive aspects of such networks and makes it easy to incorporate security and privacy proposals from the research community. To the best of our knowledge, there is no simulator that models privacy and security aspects of VANETs adequately. Existing simulators tend to focus on the simulation of the network layer, which is certainly a vital topic of VANET research in general. Although VANETs inherit the security issues of other wireless networks, security threats that exploit the specific properties of the vehicular setting originate on the application layer. This is why VANETsim focuses on a simulation of application layer protocols.

III. VANETSIM OVERVIEW

VANETsim is a lightweight, open source (GNU GPLv3), discrete event traffic and communications simulator that focuses on the analysis of security concepts in VANETs. It consists of ≈26,500 lines of platform-independent Java code. It concentrates on simulation of the *application layer* to reach high performance. VANETsim contains four main components: a *GUI*, the *Scenario Creator*, the *Simulation Core*, and the *Post Processing Engine* (cf. Fig. 1).

The **GUI** provides a graphical map editor that allows the investigator to create and manipulate road maps. Maps can either be created from scratch or imported from *OpenStreetMap* and are stored in XML files, which facilitates interoperability with other tools. Further, the GUI visualizes the simulation process in an interactive, zoomable map that displays both roads and vehicles. As an alternative to the GUI mode the simulation can be executed on the command line to run multiple experiments in a batch (faster than real time).

The **Scenario Creator** offers the ability to rapidly prepare a series of experiments in which individual parameters are varied automatically. Experiments are saved in scenario files (XML) and can be released by researchers together with scientific papers to ensure easy repeatability (*open research*).

The **Simulation Core** carries out the actual simulation. It has access to the *map*, including all infrastructure, vehicles and security/privacy concepts relevant for the simulation. Vehicles operate with a *Traffic Model*, derived from well-known models (cf. [24]–[26]) and navigate to individually determined destinations routed by the A* algorithm [27], [28]. Communication of vehicles is simulated distinguishing between so-called *beacons* and *special-purpose messages*, the two message types typically encountered in VANETs (cf. [1], [29]).

We have implemented a number of **privacy concepts** for VANETs that have been published so far, among them Mix Zones [2], [3], Silent Periods [4], [6], SLOW [7] and Pro-Mix [8]. We have also started to integrate **security concepts** into the simulator (cf. [9] and [30]). Based on our experiences with these concepts we believe that it will be quite easy to implement and adopt additional concepts in VANETsim in the future.

In the last component, the **Post Processing Engine**, the log files created during the simulation are analyzed and processed to create tables, charts or to visualize results obtained during the simulation right on the interactive map of the GUI.

IV. SIMULATION CORE

The Simulation Core is the centerpiece of VANETsim. It consists of five sub-components, which will be described in the following sections.

A. The Map

A *map* is a rectangular two-dimensional area with a Cartesian coordinate system that identifies every point with a resolution of 1.0 *cm*. Maps contain various elements, e. g., streets and vehicles, which have to be accessed during the simulation with low latency. Fast access is achieved by a two-step search process. The map is divided into several (rectangular and disjoint) *tiles* that cover equally sized areas. In the first step the location is approximated by determining the respective tile (using fast integer divisions). In the second step the exact location is determined by enumerating over all elements located within the tile. The road network is modeled with a graph-like data structure consisting of *nodes* (e. g., for intersections) and *streets* (cf. the model used by *OpenStreetMap* [31]).

VANETsim supports both, importing maps of real cities in order to study the real-world behavior of a technique, as well as creating road networks from scratch in order to provoke special traffic situations or to verify analytical results. We opted to work with maps from *OpenStreetMap* [31], because of the large amount of freely available high-quality material. During import the plain road network as well as meta information is extracted, including street names, speed limits, the number of lanes and their directions. We also have included

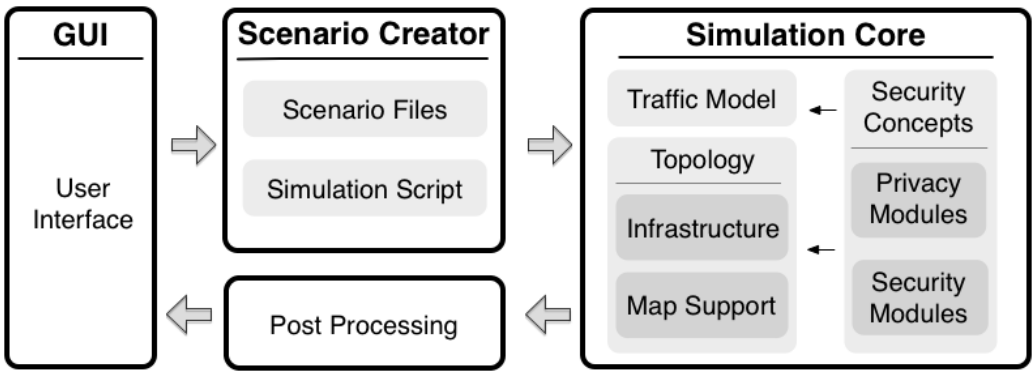

Figure 1. Architecture of the VANETsim platform

the option to import POIs (Points of interest, e. g., schools, hospitals and police stations), which can be used as waypoints in the simulation or to adapt the behavior of drivers in certain areas.

As an alternative to imported *maps* from *OpenStreetMap*, road networks may also be built from scratch with the integrated visual map editor. Of course, changing imported *maps* is also possible. The editor supports the user with few dialog options and allows the placement of roads by simply clicking on the map. Changing the attributes of road segments is possible as well as deleting them. All data may be saved in XML files for later reuse.

B. Infrastructure

Some VANET architectures, e. g., [1], envision so-called Road-side Units (*RSU*s), fixed infrastructure installations that increase the robustness of communications in the ad-hoc network by forwarding messages. So even if there is a big gap between vehicles, an *RSU* might enable communication between them. RSUs can be distributed on the map and their transmission range can be configured individually.

VANETsim also allows to deploy RSUs that provide additional security mechanisms for vehicles, e. g., Pro-Mix Zones [8]. Moreover, it facilitates the simulation of the effectiveness of special attacks, in which an adversary has gained control over single or multiple RSUs (Attacker Road-side-Units).

C. Vehicular Traffic Model

VANETsim simulates vehicular traffic on two levels, the first one determines movement and behavior of vehicles on a microscopic level in a given situation, while the second one is responsible for macroscopic decisions (navigation and routing).

Accurate simulation of **individual vehicles** is of great importance in order to approximate real-world behavior. VANETsim uses a **microscopic model** of traffic flow [25]. Each vehicle is simulated individually and the simulated drivers make their own decisions based on simulated traits and their personal view of a traffic situation. In each simulated epoch, the movement of a vehicle is influenced in five consecutive steps: (1) The vehicle checks, whether it has to reduce its speed (e. g., because of an obstacle or when approaching an intersection). (2) The vehicle changes its lane (if possible), if it would have to slow down on the current lane and can avoid this on the new lane. (3) The vehicle adjusts its speed. If there is no obstacle, it tries to travel as fast as possible (while observing its maximum speed and the road's speed limit). (4) The vehicle's velocity is reduced by a random amount to simulate imperfect driving. (5) The vehicle is moved according to its speed and the duration of the simulation step.

Our simulation runs are executed in discreet steps to be able to reduce problems with concurrency during the computations. However, we use continuous functions so that the time slices of the simulation steps can be adjusted at will. Our default time interval for one step is $40\,ms$ (of simulated time).

Each vehicle has several different attributes, for instance its maximum velocity, acceleration and deceleration rate and an attribute that represents the driver's politeness. These values may be different for all vehicles. As no traffic model suited our requirements perfectly we derived a new microscopic model from the well-investigated traffic models by Nagel and Schreckenberg [24], Krauß [25] and Treiber et. al [26]. The key features of our model are high performance, multilane support, recognition of traffic signals, road blockings and a certain degree of unpredictability to model the "human factor".

Apart from exhibiting realistic behavior on a microscopic level, the movement of vehicles is also modeled on a macroscopic level in VANETsim (**navigation and vehicle routing**). At its creation, each vehicle receives a list of at least two waypoints (start, destination, and, if applicable, any number of intermediate waypoints). The waypoints are represented by their x and y coordinates on the map as well as a road identifier, which allows to differentiate between overlapping streets (e. g., due to bridges).

The routing algorithm is performed once at the creation of the vehicle and whenever new incoming information (e. g., about a traffic jam) triggers a new route calculation. The vehicle begins its journey at the starting point and travels along

the waypoints to the destination. We assume that all drivers choose ideal routes (e. g., all of them are equipped with a GPS) using the A* algorithm [27], [28], which is a common choice because it is able to find the optimal route (if existing) without having to exhaustively evaluate all possible routes. The algorithm tries to optimize the travel time based on the length of the street segments and any speed limits on them.

D. Vehicle Communications

The simulation of *communication* between vehicles is as important as the simulation of their movements. VANETsim builds on the concepts described in [1] and [29]: It supports the transmission of both *beacons* and *special-purpose messages*, the two concepts for inter-vehicular communication assumed in most (security-related) VANET publications. The transmission range of each vehicle is set individually. As VANETsim focuses on the application layer, the lower protocol layers as well as physical characteristics of the propagation of radio waves are excluded from the simulation. Furthermore, no real encryption of messages is performed, instead, messages contain a boolean attribute that indicates whether they are encrypted or not. VANETsim assumes that adversaries cannot break the employed cryptographic primitives. Therefore, the contents of encrypted messages are not disclosed to them.

Beacons [32] (sometimes also referred to as "heartbeat" or "general safety messages" [33]), are messages that are broadcast regularly and frequently by all vehicles. The interval between two *beacons* can be globally configured for each simulated scenario. *Beacons* contain an identifier of the sender as well as its location and speed. To resemble behavior in reality, message transmission is not synchronized, i. e., once a vehicle is spawned, it waits a random amount of time t before it broadcasts its first *beacon*. To ensure that simulations can be reproduced, t is a pseudorandom value derived from the starting position of the respective vehicle.

Besides *beacons* VANETsim supports four **special-purpose message types**: "Stopped/Slow Vehicle Advisor", "Road Hazard Condition Notification", "Emergency Electronic Brake Lights" and "Emergency Vehicle Approaching" (cf. [29]). Those messages are transmitted whenever an event occurs that requires to inform other VANET participants. In contrast to *beacons*, which are not forwarded by one's peers, *special-purpose messages* are forwarded in order to inform other vehicles about an emergency situation. Depending on the type of message, VANETsim supports various forwarding protocols, e. g., *Cached Greedy GeoCast* [34], and reactions to the messages. Apart from that, VANETsim also supports the simulation of attackers that fake *special-purpose messages* and the GUI allows the user to configure the occurrence of emergency situations in order to force the transmission of *special-purpose messages*.

TABLE I
SECURITY CONCEPTS IMPLEMENTED AS MODULES IN VANETSIM

Name	Type	Literature
Silent Period	Privacy	[4], [6]
SLOW	Privacy	[7]
Mix Zones	Privacy	[2], [3]
Pro-Mix	Privacy	[8]
Ruj. et al.	IDS	[9]
REST-Net	IDS	[30]

E. Security and Privacy Modules

As the main purpose of VANETsim is to support researchers in the area of VANET security, the Security and Privacy Modules are of special interest. The clearly structured software architecture of VANETsim provides an easy way to compose multiple existing security concepts as well as implement new ones. Security and Privacy Modules are either executed by vehicles on their own or they are part of the infrastructure on the map.

As *beacons* and *special-purpose messages* in VANETs may disclose data like ID, speed, acceleration and the location to other vehicles, privacy issues arise. The **Privacy Modules** of VANETsim contain ready-to-use implementations of a number of vehicle-enabled privacy concepts, e. g., Silent Periods [4], [6], SLOW [7], as well as infrastructure-enabled privacy concepts, e. g., Mix Zones [2], [3], Pro-Mix [8]. All these concepts aim to provide a secure, unlinkable pseudonym switchover via radio silence or encryption. As shown in [35] simply changing pseudonyms is not sufficient due to the high frequency of *beacons* (e. g., every 200 *ms*). In Silent Periods all vehicles synchronously stop sending *beacons*, resulting in a period of radio silence at a specific time interval. In Mix Zones the radio silence (or the encrypted communication in the case of Pro-Mix) is tied to a specific location (e. g., a junction). In SLOW a vehicle enters radio silence when driving less than a pre-defined speed (e. g., 30 *km/h*). VANETsim can also be used to simulate attacks against these concepts. Table I offers an overview of the concepts that have been implemented so far. An empirical comparison of the effectiveness of these privacy concepts with VANETsim and a description of attacks can be found in [10]. The results provide insight into the dependencies that exist between the effectiveness of the privacy concepts and various parameters (e. g., road structure, vehicle density). In future real-world applications a specific situation (with a individual set of parameters) might demand a explicit privacy concept to reach the best protection for the user. VANETsim allows researchers to analyze the influence of the aforementioned and other parameters (e. g., aggressive vs. polite driving, structured vs. obscure street layouts) on different privacy concepts to select the best solution for a situation.

Besides privacy concerns, the manipulation of messages in VANETs is seen as another major security threat. Various Intrusion Detection Systems (IDS) have been proposed to

protect against the injection of false data into Vehicular Ad Hoc Networks [9], [36], [37]. IDS typically analyze network traffic to check for signatures or statistical anomalies. In VANETs the detection of false data through network traffic is challenging. An attacker can send correct *special-purpose messages* as a normal vehicle (inside attacker) because it is almost impossible to achieve veracity of the sensors [38]. To detect false data in VANETs Intrusion Detection Systems often use application layer data such as position, time and application-specific information to conduct a form of context verification [39].

VANETsim supports implementing such techniques as **Security Modules** and to test them using attacker vehicles that send faked *special-purpose messages*. Currently, there is a general-purpose IDS module from which two rule-based implementations have been derived so far (cf. Table I for an overview, [30] for an empirical comparison). VANETsim can help researchers identify the most promising IDS solutions for VANETs. While few concepts (c.f [36]) already provide simple simulation-based evaluations in a specific setting, due to own implementations there is no way to compare them to each other. Furthermore, like with privacy concepts, the influence of different simulation parameters (e. g., amount of vehicles with WiFi, vehicle density, driving style) on the IDS has not been examined in the literature sufficiently. General simulation data and data of specific traffic situations might also be used to train rule-based IDS or systems that employ machine learning techniques.

Apart from the possibilty to analyze specific security concepts, VANETsim enables users to analyze collaboration and interdependencies between privacy concepts, IDS or both. The use of different privacy concepts in specific traffic situations might result in a better level of privacy, but it might also lead to new identifiable patterns (e. g., a vehicle changes its pseudonym more frequently due to a specific driving style). For IDS the utilization of more than one system might improve detection results, but a efficient way to reach a common decision has to be investigated. The interdependencies of both security concepts might also be of interest. Privacy concepts apply radio silence and therefore cause a loss of *beacons* for a specific time. IDS require those *beacons* to evaluate *special-purpose messages* using context verification. VANETsim offers a platform to analyze these possible collaborations as well as interdependencies and find applicable solutions or tradeoffs.

V. SCENARIO CREATOR AND POST PROCESSING ENGINE

The *Scenario Creator* of VANETsim supports the researcher in designing and running experiments in an easily repeatable way. In the GUI, the user can define ranges for individual parameters for vehicles (e. g., speed, acceleration, politeness, WiFi support, WiFi range, etc.), security concepts (e. g., Mix Zone radius, Silent Period duration, IDS thresholds, etc.) and attacks (e. g., thresholds, attack types, etc.) as well as an iteration count to obtain significant results by aggregating multiple consecutive runs. The output of the Scenario Creator is a scenario file and a ready-to-use script to start all simulations in Console Mode.

The analysis of the results of a simulation is facilitated by the *Post Processing Engine*, which provides various scripts to aggregate the details from the event log. Moreover, the aggregated results can be graphed with *gnuplot* scripts to allow for visual comparison. Apart from being aggregated into an overall result, which is useful to determine the expected effectiveness of attacks on the deployed security and privacy concepts, the security events are also fed back into the map: During and after the simulation, all entities can be inspected within the interactive map to gain insights on a local level, e. g., in order to analyze the utility of a Mix Zone at an intersection.

In order to give an impression of the graphical representation of simulation results fed back to the GUI, we briefly illustrate the currently implemented visualization for Mix Zones, a privacy concept proposed previously [2], [3]. The goal of a Mix Zone is to prevent an attacker from tracking a vehicle by collecting its *beacons* (which is easy due to the vehicle-specific pseudonym contained in them). Once a vehicle enters a Mix Zone it stops to broadcast *beacons* and switches to a new pseudonym in order to provide for unlinkability. From the perspective of the adversary the Mix Zone is a black box. He can only log which vehicles enter and which leave a Mix Zone and try to link the respective pseudonyms. Figure 2 shows the results of an attack on two Mix Zones (in grey). The red and green circles represent the different entrances into the Mix Zone, with the circle radius indicating the amount of vehicles entering the zone at this entrance. Green circles show secure entrances, i. e., the adversary failed to track vehicles that entered the zone at this entrance. Red circles indicate insecure entrances, i. e., due to insufficient vehicular "cover traffic" or predictable routes the adversary could establish an association between the pseudonyms used by the vehicles before entering and after leaving the Mix Zone. This visualization helps to determine in which scenarios Mix Zones are suitable and when to prefer other concepts for privacy protection. The aforementioned scenario illustrates a major benefit of a simulation-based evaluation as made available by VANETsim in comparison to using analytical models. With a single simulation run in a real city containing multiple Mix Zones a user can examine numerous traffic situations. This helps to detect weak spots of a security concept regarding specific traffic parameters. In contrast, an analytical analysis of all possible parameters (e. g., the road structure, driving styles, nearby hospitals) is typically not feasible.

Even more detailed analysis might be accomplished by displaying this Post Processing results while re-running this exact same simulation. Feeding back simulation results into the map can lead to new insights regarding attacks and security concepts. A further example of the visual capabilities enabled by Post Processing is given in Fig. 3.

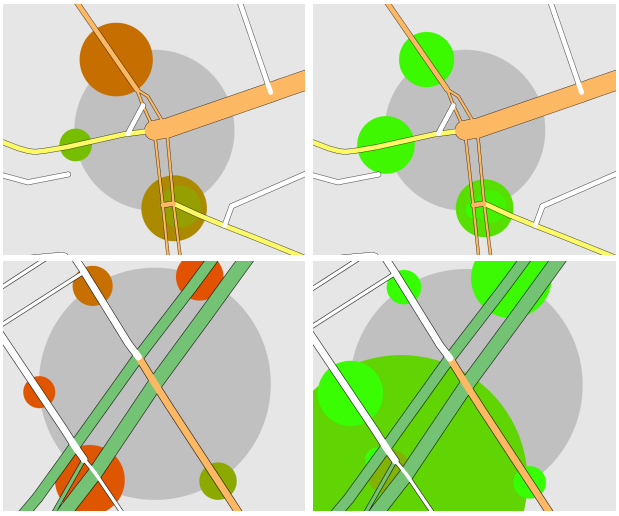

Figure 2. Post-processing and visual feedback for graphical analysis of an attack on (top, left to right) a Mix Zone in Berlin (1250 vehicles and 20,000 vehicles) and New York (bottom) (1250 vehicles and 20,000 vehicles), cf. [10].

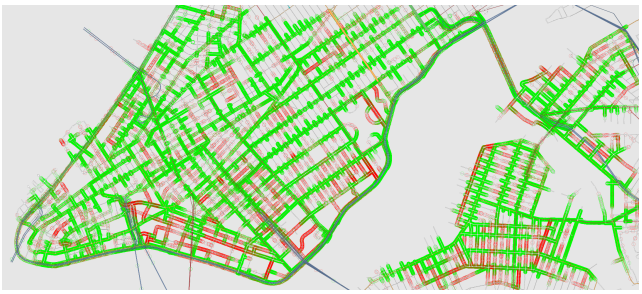

Figure 3. Result of Post Processing a simulated attack on Silent Periods in Manhattan with 20,000 vehicles; successfully linked pseudonyms are displayed in red color [10].

VI. GUI

The Graphical User Interface (GUI) provides facilities to configure a simulation and interact with it while it is running. Moreover, it contains a zoomable and scrollable visualization of the map, which displays the road network as well as all vehicles and infrastructure elements like RSUs (see Figs. 4 and 5). Additional information overlays, e. g., indicating the transmission range of vehicles, can be activated as needed. Elements on the map can be clicked for closer inspection via the side bar, which contains all relevant meta data, such as street names, speed limits and the current speed of a vehicle.

The map and scenario editor is tightly integrated into the GUI. Street segments can be configured in the sidebar and placed by clicking on the *map*. Vehicles and other dynamic objects can be also placed by manually setting their waypoints on the map. Alternatively, large numbers of vehicles can be automatically generated and distributed randomly.

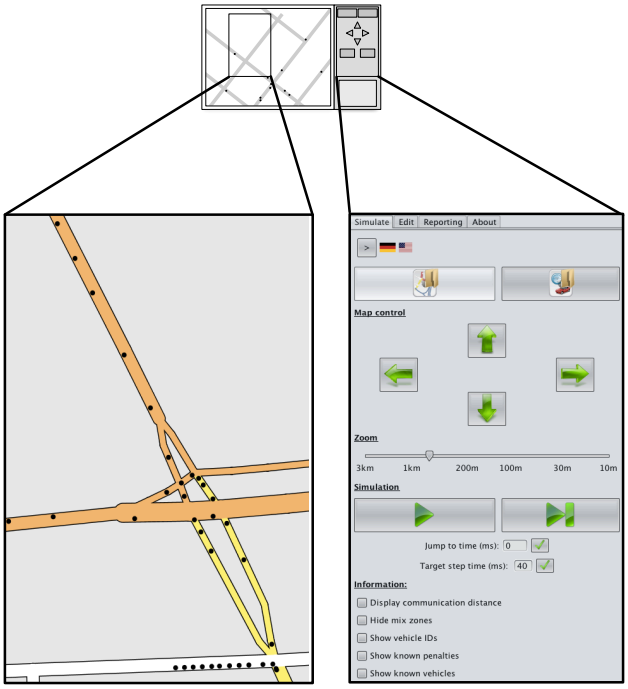

Figure 4. Graphical User Interface

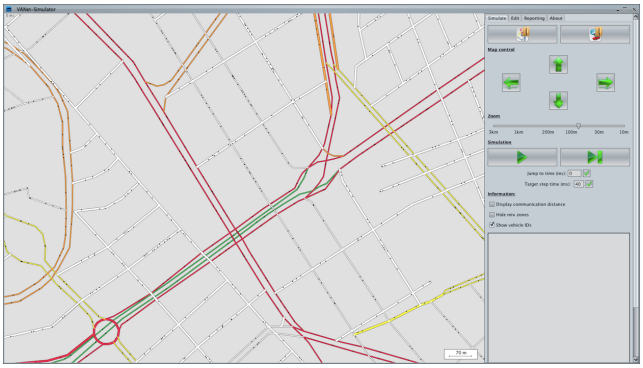

Figure 5. Simulation run on a map of Puebla

VII. PERFORMANCE EVALUATION

A central objective guiding the development efforts related to VANETsim is high performance on off-the-shelf hardware. In the following, we provide results from typical simulation runs using different maps and different numbers of vehicles to give an impression of the attainable scalability. All simulations were performed on a desktop computer ("Intel Core i5-3470 3.20 Ghz" quad-core CPU, 8 GB RAM) running Ubuntu 13.04 64 Bit and VANETsim Version 1.0. The maps for the evaluation are *OpenStreetMap* exports of the inner cities of Berlin (Germany), New York City (USA) and Puebla (Mexico) with a total street length of 854, 547 and 496 *km* respectively. This concrete choice is also motivated by the different street layout, with Berlin having grown organically, Manhattan in New York City being planned accurately, and Puebla containing a mixture of layout elements.

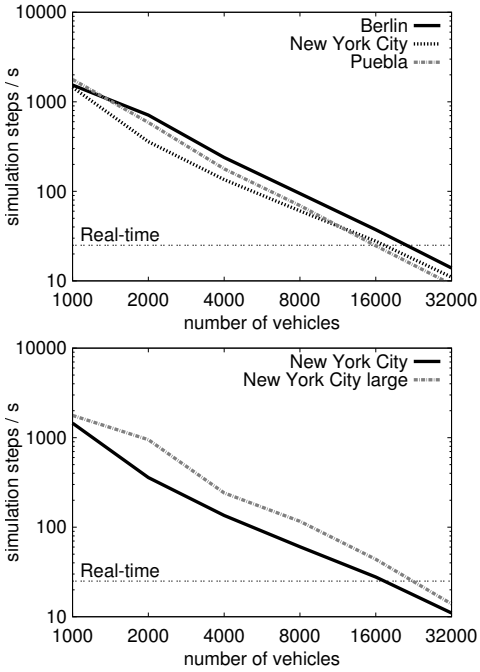

Figure 6. Performance analysis of VANETsim

The scenario was configured with a general-purpose configuration: The *beacon* interval was set to 240 *ms*, the vehicular WiFi range to 100 *m*, each simulation step took 40 *ms* simulated time. Vehicles were configured to immediately react to incoming messages that indicate a change in the traffic situation en route by recalculating their route (cf. Sect. IV-C). Each simulation run covered 10,000,000 ms of simulated time (≈2.7 h) and was carried out five times, each time with differing, randomly created waypoints. We measured the total runtime of each simulation (250,000 simulation steps) and determined the average number of simulation steps per second. The results are shown in Fig. 6 (left-hand side). The number of processed simulation steps per second decreases with increasing vehicle density logarithmically. The realtime threshold is crossed at about 16,000 vehicles for each city. Higher numbers of vehicles *can* be simulated, but not in real time (25 simulation steps per second at a step size of 40 *ms*) on our machine.

The length and layout of the road network has a moderate impact on the achievable performance. However, surprisingly, the largest road network with the highest number of nodes (Berlin) performs best. At a second glance, the larger network results in a wider distribution of vehicles and therefore reduces the communication overhead, which is mainly determined by the number of reachable neighbors that have to be maintained for each vehicle. This suggests that the attainable performance is less dependent on the absolute number of vehicles, but rather a function of the density of traffic. In order to confirm this conjecture we repeated the experiment with a much larger map of New York City with a total street length of 2292 *km*. The results (cf. Fig. 6, right-hand side) indicate that our hypothesis is correct: The performance of the simulation is significantly better on the large map, even though the number of static elements is considerably higher.

VIII. CONCLUSION

In this paper we present the design of VANETsim, an open source simulator specifically tailored for empirical analysis of privacy and security concepts in VANETs. The current version provides ready-to-use implementations of numerous concepts from the literature and the flexible architecture ensures that future concepts can be integrated easily. The Simulation Creator supports researchers in building repeatable experiments, which allows them to verify security concepts on artificially crafted as well as real-world maps provided by the *OpenStreetMap* project. VANETsim employs a microscopic vehicular traffic model drawing from state-of-the art research in the field of traffic engineering. The Graphical User Interface supports the user during both phases, experimental design as well as evaluation, feeding back the results obtained during the simulation. A console mode simplifies unattended batch mode processing to gather a large number of results. In our experiments, VANETsim was able to simulate more than 16,000 vehicles and their communications in real-time on off-the-shelf desktop hardware.

We believe that with its feature set, VANETsim fills a gap in the landscape of simulation tools that are currently available for researching privacy and security risks as well as countermeasures in vehicular networks. VANETsim may also prove as a suitable tool for teaching these concepts to students, which became apparent recently, when we were approached by students that use VANETsim for their coursework. Therefore, in future work we will not only include and evaluate additional security concepts, but also extend VANETsim with a descriptive presentation mode that can be used to teach the functionality of security and privacy concepts in a hands-on manner.

ACKNOWLEDGMENT

The authors would like to thank Bernhard Gruber who implemented the first working prototype of VANETsim in 2008.

REFERENCES

[1] K. Plößl and H. Federrath, "A privacy aware and efficient security infrastructure for vehicular ad hoc networks," in *Security in Information Systems: Proceedings of the 5th International Workshop on Security in Information Systems – WOSIS 2007*, 2007.

[2] A. R. Beresford and F. Stajano, "Location privacy in pervasive computing," *IEEE Pervasive Computing*, vol. 2, 2003.

[3] ——, "Mix zones: User privacy in location-aware services," in *2nd IEEE Conference on Pervasive Computing and Communications Workshops (PerCom 2004 Workshops), 14-17 March 2004, Orlando, FL, USA*. IEEE Computer Society, 2004, pp. 127–131.

[4] L. Huang, K. Matsuura, H. Yamane, and K. Sezaki, "Enhancing wireless location privacy using silent period," in *Proceedings of the 2005 IEEE Wireless Communications and Networking Conference (WCNC)*, New Orleans, LA, USA, March 2005, pp. 1187–1192.

[5] M. Li, K. Sampigethaya, L. Huang, and R. Poovendran, "Swing & swap: user-centric approaches towards maximizing location privacy," in *Proceedings of the 2006 ACM Workshop on Privacy in the Electronic Society, WPES 2006, Alexandria, VA, USA, October 30, 2006*. ACM, 2006, pp. 19–28.

[6] L. Huang, H. Yamane, K. Matsuura, and K. Sezaki, "Silent cascade: Enhancing location privacy without communication qos degradation," in *Security in Pervasive Computing, Third International Conference, SPC 2006, York, UK, April 18-21, 2006, Proceedings*, ser. Lecture Notes in Computer Science, vol. 3934. Springer, 2006, pp. 165–180.

[7] L. Buttyán, T. Holczer, A. Weimerskirch, and W. Whyte, "SLOW: A practical pseudonym changing scheme for location privacy in VANETs," in *Proceedings of the IEEE Vehicular Networking Conference (VNC), Tokyo, Japan, 2009*. IEEE, October 2009.

[8] F. Scheuer, K.-P. Fuchs, and H. Federrath, "A Safety-Preserving Mix Zone for VANETs," in *Proceedings of Trust, Privacy and Security in Digital Business - 8th International Conference, TrustBus 2011, Toulouse, France*, ser. Lecture Notes in Computer Science. Springer Berlin / Heidelberg, 2011, vol. 6863, pp. 37–48.

[9] S. Ruj, M. A. Cavenaghi, Z. Huang, A. Nayak, and I. Stojmenovic, "Data-centric misbehavior detection in vanets," *CoRR*, vol. abs/1103.2404, 2011.

[10] A. Tomandl, F. Scheuer, and H. Federrath, "Simulation-based evaluation of techniques for privacy protection in VANETs," in *Proceedings of the IEEE International Conference on Wireless and Mobile Computing, Networking and Communications (WiMob'2012)*, 2012.

[11] M. Piórkowski, M. Raya, A. L. Lugo, P. Papadimitratos, M. Grossglauser, and J.-P. Hubaux, "TraNS: realistic joint traffic and network simulator for VANETs," *ACM SIGMOBILE Mobile Computing and Communications Review*, vol. 12, no. 1, pp. 31–33, 2008.

[12] C. Sommer, Z. Yao, R. German, and F. Dressler, "On the Need for Bidirectional Coupling of Road Traffic Microsimulation and Network Simulation," in *9th ACM International Symposium on Mobile Ad Hoc Networking and Computing (Mobihoc 2008): 1st ACM International Workshop on Mobility Models for Networking Research (MobilityModels 2008)*. Hong Kong, China: ACM, Mai 2008, pp. 41–48.

[13] C. Sommer and F. Dressler, "Progressing Towards Realistic Mobility Models in VANET Simulations," *IEEE Communications Magazine*, vol. 46, no. 11, pp. 132–137, November 2008.

[14] C. Sommer, R. German, and F. Dressler, "Bidirectionally Coupled Network and Road Traffic Simulation for Improved IVC Analysis," *IEEE Transactions on Mobile Computing*, vol. 10, no. 1, pp. 3–15, Januar 2011.

[15] J. Härri, F. Filali, C. Bonnet, and M. Fiore, "VanetMobiSim: generating realistic mobility patterns for VANETs," in *Proceedings of the Third International Workshop on Vehicular Ad Hoc Networks, VANET 2006*. Los Angeles, CA, USA: ACM, September 2006, pp. 96–97.

[16] M. Fiore, J. Härri, F. Filali, and C. Bonnet, "Vehicular mobility simulation for vanets," in *Proceedings 40th Annual Simulation Symposium (ANSS-40 2007)*. Norfolk, Virginia, USA: IEEE Computer Society, 2007, pp. 301–309.

[17] J. Härri, M. Fiore, F. Filali, and C. Bonnet, "Vehicular mobility simulation with VanetMobiSim," *SIMULATION: Transactions of The Society for Modeling and Simulation International*, vol. 87, no. 4, April 2011.

[18] M. Behrisch, L. Bieker, J. Erdmann, and D. Krajzewicz, "SUMO - Simulation of Urban MObility - an Overview," in *Proceedings of the Third International Conference on Advances in System Simulation, SIMUL 2011*, Barcelona, Spain, Oktober 2011, pp. 63–68.

[19] G. Pongor, "OMNeT: Objective Modular Network Testbed," in *Proceedings of the International Workshop on Modeling, Analysis, and Simulation On Computer and Telecommunication Systems, MASCOTS '93*. San Diego, CA, USA: The Society for Computer Simulation International, 1993, pp. 323–326.

[20] A. Varga, "The OMNeT++ Discrete Event Simulation System," *Proceedings of the European Simulation Multiconference (ESM'2001)*, Juni 2001.

[21] R. Mangharam, J. Meyers, D. Rajkumar, and D. D. Stancil, "A multi-hop mobile networking test-bed for telematics," in *Society for Automotive Engineers (SAE) World Congress*, 2005.

[22] R. Mangharam, D. S. Weller, D. D. Stancil, R. Rajkumar, and J. S. Parikh, "Groovesim: A topography-accurate simulator for geographic routing in vehicular networks," in *Proceedings of Second ACM International Workshop on Vehicular Ad hoc Networks (Mobicom/VANET 2005)*, 2005.

[23] R. Mangharam, D. S. Weller, R. Rajkumar, P. Mudalige, and F. Bai, "Groovenet: A hybrid simulator for vehicle-to-vehicle networks," in *Second International Workshop on Vehicle-to-Vehicle Communications (V2VCOM), San Jose, USA*, 2006.

[24] K. Nagel and M. Schreckenberg, "A cellular automaton model for freeway traffic," *J. Phys. I France*, vol. 2, no. 12, pp. 2221–2229, September 1992.

[25] S. Krauß, "Microscopic Modeling of Traffic Flow: Investigation of Collision Free Vehicle Dynamics," Dissertation, Universität Köln, Köln, April 1998.

[26] M. Treiber, A. Hennecke, and D. Helbing, "Congested traffic states in empirical observations and microscopic simulations," *Physical Review E*, vol. 62, pp. 1805–1824, August 2000.

[27] N. J. Nilsson, *Principles of Artificial Intelligence*, ser. Symbolic Computation: Artificial Intelligence. Springer, Mai 1982.

[28] A. Stentz, "Optimal and Efficient Path Planning for Partially-Known Environments," in *Proceedings of the International Conference on Robotics and Automation, ICRA94*, 1994, pp. 3310–3317.

[29] F. Bai, H. Krishnan, V. Sadekar, G. Holl, and T. Elbatt, "Towards characterizing and classifying communication-based automotive applications from a wireless networking perspective," in *Proceedings of IEEE Workshop on Automotive Networking and Applications (AutoNet)*, 2006.

[30] A. Tomandl, K.-P. Fuchs, and H. Federrath, "REST-Net: A rule-based IDS for VANETs," *to be published at Wireless and Mobile Networking Conference (WMNC 2014)*, 2014.

[31] M. M. Haklay and P. Weber, "OpenStreetMap: User-Generated Street Maps," *IEEE Pervasive Computing*, vol. 7, no. 4, pp. 12–18, 2008.

[32] L. Buttyán, T. Holczer, and I. Vajda, "On the effectiveness of changing pseudonyms to provide location privacy in vanets," in *Proceedings of the Fourth European Workshop on Security and Privacy in Ad hoc and Sensor Networks (ESAS 2007)*, Juli 2007.

[33] M. Raya and J.-P. Hubaux, "The security of vehicular ad hoc networks," in *Proceedings of the 3rd ACM workshop on Security of ad hoc and sensor networks (SASN)*. ACM, November 2005, pp. 11–21.

[34] C. Maihöfer, R. Eberhardt, and E. Schoch, "CGGC: Cached Greedy Geocast," in *Wired/Wireless Internet Communications*. Springer Berlin / Heidelberg, 2004, vol. 2957, pp. 171–182.

[35] B. Wiedersheim, Z. Ma, F. Kargl, and P. Papadimitratos, "Privacy in inter-vehicular networks: Why simple pseudonym change is not enough," in *Wireless On-demand Network Systems and Services (WONS), 2010 Seventh International Conference on*. IEEE, 2010, pp. 176–183.

[36] M. Ghosh, A. Varghese, A. A. Kherani, and A. Gupta, "Distributed misbehavior detection in vanets," in *Wireless Communications and Networking Conference, 2009. WCNC 2009. IEEE*, 2009, pp. 1–6.

[37] M. Raya, P. Papadimitratos, I. Aad, D. Jungels, and J.-P. Hubaux, "Eviction of misbehaving and faulty nodes in vehicular networks," *IEEE Journal on Selected Areas in Communications*, 2007.

[38] D. Gollmann, "Veracity, plausibility, and reputation," in *Information Security Theory and Practice. Security, Privacy and Trust in Computing Systems and Ambient Intelligent Ecosystems*. Springer, 2012, pp. 20–28.

[39] T. Leinmüller, E. Schoch, and C. Maihöfer, "Security Requirements and Solution Concepts in Vehicular Ad Hoc Networks," *Fourth Annual Conference on Wireless on Demand Network Systems and Services*, 2009.