The Virtual Power System Testbed and Inter-Testbed Integration

David C. Bergman, Dong Jin, David M. Nicol, and Tim Yardley

University of Illinois at Urbana-Champaign
{dcbergma, dongjin2, dmnicol, yardley}@illinois.edu

Abstract

The Virtual Power System Testbed (VPST) at University of Illinois at Urbana-Champaign is part of the Trustworthy Cyber Infrastructure for the Power Grid (TCIP) and is maintained by members of the Information Trust Institute (ITI). VPST is designed to be integrated with other testbeds across the country to explore performance and security of Supervisory Control And Data Acquisition (SCADA) protocols and equipment. We discuss potential use cases in order to motivate the integration of VPST with other testbeds, identify requirements of interconnected testbeds, and describe our design for integration with VPST.

1. Introduction

As part of the NSF/DOE/DHS supported TCIP project [12], VPST combines large-scale simulation/emulation of networks of SCADA power devices with real power system hardware and software, and a commercial electric flow generation and distribution simulator. VPST is unique in its integration of virtual and physical equipment, as discussed in [9]. VPST's principal role so far has been to demonstrate the feasibility of integrating the components it has, the feasibility of a number of cyber-attacks, and studies of performance and effectiveness of certain other TCIP developed technologies. It is not yet to the level of general purpose utility achieved by, say, DETER, although we are working to bring it to that state. As we do so, we have redesigned the VPST architecture to support integration with other testbeds, and have already demonstrated basic integrative functionality. We aim towards enabling one to leverage resources from external collaborators and accomplish tasks that may not have been feasible without them. For example, one lab may have expertise in cyber warfare and another may emphasize actual SCADA devices. VPST provides detailed models of SCADA-specific protocols, enabling studies where an attack is mounted on a large-scale SCADA network. Such integration has a unique set of requirements, and we have extended VPST with these in mind.

This paper describes VPST and its components through the lens of three potential use case scenarios, providing some concreteness behind VPST design decisions. First, we will discuss the motivation for our system in Section 2. Section 3 then details three use cases, after which we discuss testbed requirements in Section 4. Section 5 gives a brief overview of VPST and then delves into the details of our extension. Section 6 discusses related projects and how they may use VPST as a national resource. Finally, we conclude in Section 7 while detailing some of our ongoing work.

2. Motivation

There is an emerging awareness of the need for security in SCADA systems, and the Department of Energy (DOE) is giving considerable attention to securing the power grid. An important approach to studying cyber-security of the power grid is through test systems, so as not to interfere with real systems. These test systems may use actual equipment in order to properly understand how the equipment will react in various situations. However, the scope of the grid makes it infeasible to create a test system anywhere near its full scale. Simulation and emulation help to alleviate this concern, as it is possible to create large-scale network simulation/emulations of models as large as regional or even national power grids.

3. Use Cases

3.1. Training and Human-in-the-loop Event Analysis

Early detection and prevention technologies are required as was learned from the massive mid-western blackout in August 2003[11], occurring largely because diverse failures caused a lack of situational awareness on the part of the operators. Had there been different detection mechanisms in place, the evolving disaster might have been recognized sooner, and corrected. Electrical system data is continuously read and stored which allows system state to be replayed on the testbed, pre-

senting operators with the same data as existed at the time of the blackout (or some other notable event). However, the testbed can provide real or simulated versions of new technology. This allows operators to react to what they *now* see (owing to the simulation), which may be different from the original incident. As different control decisions are made (e.g. open a breaker) the testbed shifts from playing back observed state information from the original event to generating a new state trajectory, based on the new control decisions. By integrating with real device/emulation testbeds, VPST has two significant benefits over a traditional power system training simulator: (1) actual hardware is in the loop, (2) the communications system is modeled, allowing a more faithful interaction.

This use case requires the testbed to ensure the following: secure connectivity for protecting sensitive information such as strategic decisions; reproducibility for event replay and analysis; scalability for large-scale power grid experiments; and flexibility and fidelity for easy construction of realistic scenarios from the operators' point of view.

3.2. Analysis of Incremental Deployment

Securing the power grid requires an overhaul of the existing infrastructure. However, the size and scope of the grid necessitates that the change be gradual, not instantaneous. When old and new technologies coexist, there is the possibility that unforeseen interactions may occur. For instance, current communications make heavy use of the DNP3 protocol[5], yet this protocol provides little in the way of security. Effort has been put into creating an extension to DNP3—DNP3 Secure Authentication (DNP3SA)[1] that addresses authentication concerns with DNP3. As this is deployed, the legacy support must be maintained. DNP3SA is just one example of many. We may also want to analyze strategies for incremental deployment of new technologies, tested against a multitude of network conditions (e.g. lossy networks, congested networks, insecure environments, while under attack, or with the inclusion of corrupted data packets) and collect related statistics (e.g. bandwidth usage, latency, dropped packets, success ratio for communications, overhead incurred, etc.).

When testing a new protocol, device, or policy, there are a few important requirements: reproducibility to ensure that any changes are a direct result of the new technology in question; high performance in that we must accurately model the scale of the real power grid; customizable in order to provide a quick turnover from one configuration to another; and high fidelity in order to guarantee that a new technology behaves the same in simulation as in the wild.

3.3. Attack Robustness Analysis

A third use case is analyzing the robustness of a design against an attack. For instance, the DETER testbed has a highly provisioned communications network that is often used to test new protocols against well-defined attack models. Also, Idaho National Labs (INL) has SCADA equipment for use in experiments. However, there is currently no safe way to launch an attack on a large-scale SCADA network – VPST provides such an avenue for testing the reliability of a SCADA network in the face of an attack. We can leverage the cyber-attack capability of DETER, while integrating the real power equipment that INL operates. VPST provides the crucial third testbed that can simulate the SCADA network at full-scale with the added benefit of causing no harm to the actual power grid.

The main requirements for such an experiment are: secure connectivity to guarantee that attacks are contained to the network under duress; reproducibility to allow repeated attacks against various defenses; and fidelity to ensure the system under attack behaves reliably. Additionally, this use case actualizes an attack, so it requires the ability to abstract attacks in order to prevent WAN links from saturating (e.g., during a ping flood).

4. Inter-Testbed Connection Requirements

4.1. Secure Connectivity

Since the integrated testbed is targeted for SCADA network security analysis, security of the testbed itself is an absolute requirement. The testbed may face threats from external cyber-attacks as well as internal malicious code, which could intentionally or accidentally gain unauthorized access to secret assets. A good security policy should enable several layers of protection including transmission security, authentication and access control, traffic isolation, intrusion detection, logging and reporting. However, existing security technologies are generally not implemented in real SCADA systems because real SCADA devices usually have limited processing capabilities, operate in real time, and are typically not designed with cyber-security in mind.

To ensure a secure cross-testbed experimental environment, access control policies and strong authentication should be enabled at all access points; the access model should deny access to any party not explicitly allowed; and only ports and services required for operations should be allowed. Open PCS Security Architecture for Interoperable Design(OPSAID)[10] is a framework for accelerating development and adoption of a wide range of security functionality in control systems using IP networks. Its technical approach is to use existing open-source technologies whenever possible. This approach could be augmented with additional layers of

protection to cover the full scope of secure connectivity.

We have analyzed the current testbed situation and have implemented various measures to ensure the secure connectivity required by VPST. First, we have segregated the virtual nodes onto a private network. This prohibits incidental traffic emanating from our network and interfering with the outside world unless specifically redirected to do so. Also, communication between all components utilizes proven secure protocols. The local systems all communicate using OpenVPN (using *SSL* for encryption). The remote testbeds connect using IPSec, which provides basic security guarantees.

4.2. Performance

For a testbed to be useful, it must provide timely results. In order for this to occur, certain characteristics must hold true for both the inter-testbed connections (ITC) as well as the performance of a single testbed. When connecting two or more testbeds, precautions must be taken in order to provide guarantees about performance, primarily concerning the latency of information transfer between testbeds. To allow for efficient scaling, our ITCs handle multiple connections by having a single point of contact and then distributing the workload to other components. Instead of having an individual connection for each emulated host, we aggregate the data and use a tagging mechanism to differentiate between the hosts using "super nodes" as discussed in Section 5.3.2.

Connecting with a simulated environment has its own set of issues, namely the interaction between emulation and simulation. The communication simulator in VPST has the capability to absorb latency caused by emulated packets. Since emulated packets take precedence over simulated packets, real communication will always remain as close to real-time as possible. *Look-ahead* is the ability to predict the amount of simulated time that could be safely advanced in one process without causing errors in other concurrent processes. Where possible, VPST-C utilizes look-ahead algorithms in order to keep the simulation running as smoothly as possible.

The other source of latency is in the simulation control plane. The main source of information provided over this link ought to be known ahead of time and therefore can be transmitted prior to the start of simulation. By shifting the bulk of control messages outside the simulation itself, we can minimize the overhead incurred by control messages.

Another performance concern is that of scalability. One of VPST's contributions to the SCADA testbed community is that we provide a highly scalable network simulator. Therefore, we must ensure that the scale of networks we can simulate is sufficient to justify the interconnection of other testbeds with VPST. VPST is capable of simulating over a million devices and the electrical simulation can handle more than one hundred thousand buses[3]. The size of a power grid that can be simulated with this much capacity may be appreciated by comparison: a city the size of Madison, WI (about a quarter of a million people) has a grid with a couple hundred buses. In order to simulate a network of this scope in real-time, we make use of the the Trusted ILLIAC[6] as discussed in Section 5.2.

4.3. Resource Allocation

Flexible configuration has been addressed for standalone simulation/emulation testbeds in [4] and [7]. Further, an integrated testbed requires an accurate resource mapping among testbeds for balancing customizability and speed. VPST takes the decentralized approach, where interfaces to other testbeds are decomposed into modules for the ease of customization. Details are discussed in Section 5.3.

VPST intelligently partitions simulation models for balancing resources and minimizing communication overhead across multiple machines. Similarly, a good mapping may minimize the number of links across heterogeneous testbeds, though it is often hard to determine if a mapping is overloaded at the initial stage of an experiment, especially when human decisions are later involved. Therefore, techniques for overloaded link detection and dynamic resource mapping optimization are desired. Successive mappings are adjusted based on the feedback from the detection system and the prior mapping, until no overload is detected, or until all physical resources are depleted.

4.4. Reproducibility

The dynamics of the real SCADA network cover a wide range of conditions including, but not limited to the size of the network, type of underlying physical medium, available bandwidth and time-varying traffic patterns. Therefore, precisely repeating the experimental conditions and reproducing entire or partial results is a property of a good testbed. Integrating with a simulation testbed enhances reproducibility, since the entire parameter space including both input and environment configuration can be fully controlled.

Reproducibility in the scope of inter-connected testbeds requires conducting experiments in a controlled and interactive manner, especially allowing human-in-the-loop decision, as discussed in Section 3.1. Experimenters are given the opportunity to tune certain model parameters online, such as link connectivity and event response mechanisms. VPST then progresses along a new experimental trajectory, which is recorded as tcpdump/libpcap traces for analysis and later reproduction.

Lastly, the testbed must be able to handle the un-

predictability of long-distance communication. For instance, if VSPT receives traffic with remote origination, the simulation must be identical from one run to the next, regardless of inter-site latency. VSPT utilizes algorithms that deal with this on a local level, but it also must be addressed in the context of inter-operating testbeds.

4.5. Fidelity

To provide high fidelity, VPST-C must be as transparent as possible. That is, real-world equipment should not be able to tell that it is communicating with a virtual host. In order to present this appearance to a real control station, for instance, issues such as latency, realistic data patterns, and accurate virtual hosts are all important aspects of VPST. Latency has been discussed in Section 4.2, but realistic data and accurate hosts both fall under the auspice of fidelity requirements.

Realistic data patterns are created through the interactions of each of the layers modeled in VPST-C. Virtual hosts are responsible for creation of such data. These virtual hosts can be as abstract as a "router" with a simple MAC layer or as detailed as an "SEL 421 Relay" with a complete DNP3/MODBUS stack and high resolution Ethernet.

Fidelity is often a counterpoint to performance in that the more accurately a host is modeled, the more computation is required. As such, the trade-off needs to be considered individually for each project that requires interaction with VPST.

5. VPST Architecture

5.1. Base System Overview

As discussed in [9], VPST is divided into three main subsystems, as seen in Figure 1:

VPST-E handles electrical simulation. The primary component here is PowerWorld which is capable of simulating large scale electrical networks at the bus level. We use this to model city-sized or larger power grids.

VPST-C handles network simulation based on RINSE[7], which provides a highly scalable virtual network that is used to model the cyber domain of the electrical grid.

VPST-R-local represents all the real devices. Any software that is run rather than simulated resides on some real device in the VPST-R-local, and is represented inside of VPST-C by a device proxy. Devices in VPST-R-local are capable of interacting with VPST-E through a converter, and VPST-C through its emulation capability.

5.2. Enhanced VPST

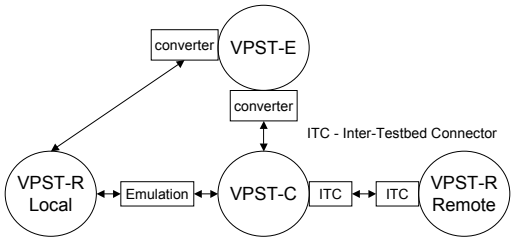
As seen in Figure 1, VPST-R-remote represents a remote test bed communicating with VPST through an ITC installed at the remote site. We have enhanced VPST with the following significant augmentations:

ITC: In order to connect with a remote testbed, we extend the basic VPST architecture by use of ITCs, discussed further in 5.3. Figures 1 and 2 both show one remote testbed, but this can be scaled to handle more than one remote connection. The ITC is responsible for ensuring that two testbeds are able to efficiently and expressively communicate with each other as to facilitate their interaction.

Trusted ILLIAC: Another enhancement to the VPST architecture we have introduced is the use of high performance computing clusters, such as the Trusted ILLIAC[6] and its 512 cores, to scale even further. VPST supports model partitioning to intelligently spread virtual nodes across the allocated processors so as to minimize inter-processor communication. This allows an almost-linear scaling of performance per core. The Local Controller is responsible for interpreting configuration files in order to partition the graph and allocate the Trusted ILLIAC resources.

Trace Files: We have also modified VPST to work with trace files. VPST now has the capability of dumping the traffic into a *tcpdump* file that can be replayed at a later time. First, this allows an external entity (e.g. a power company) to use our testbed to examine their network traces for whatever purpose they may have (e.g. testing a prevention policy). Secondly, this allows replaying experiments in order to examine how real-world equipment interacts under different configurations.

Figure 1. VPST System Diagram



5.3. Inter-Testbed Connector Framework

Figure 2 shows the ITC architecture. The control plane and the data plane are separated because the control channel is often more sensitive to latency and also requires less bandwidth than the data channel. In addition, control signals likely require higher security level than traffic data does. When two testbeds are initially connecting to each other, the ITC in VPST-C is always identified as the master controller, which is responsible for making a high-level resource allocation plan based on the requests initiated from slave ITCs in other testbeds. Only the local ITC is allowed to communicate with the local controller so as not to violate the local security policy. The architecture design is decomposed

into individual modules to allow customization and facilitate the extension of functionality.

5.3.1. Simulation Control Plane

ITC Controller, the hub of an ITC, exchanges control commands with a remote ITC and collects/distributes these control commands with the local control plane. *Secure connectivity* is ensured here through access control policies and authentication mechanisms.

Resource Allocator is responsible for managing *resource allocation* based on requests from remote testbeds and responses from local resources. Load balancing should be optimized here to improve performance and fidelity. The Resource Allocator is also responsible for checking the correctness of the topology mapping and detecting IP address conflicts across multiple testbeds. Since VPST-C is the central component of the system and provides the interconnection, IP uniqueness should be guaranteed by VPST-C. IP conflicts in the simulator are automatically resolved through its IP reassignment procedure in the case of simulation, and performs IP translation when real devices are used.

Resource Configurator is responsible for configuring network components such as hosts, links and traffic. Using the Domain Modeling Language (DML) in VPST-C makes the configuration process both flexible and *reproducible*. DML is a simple scripting language with a hierarchical attribute tree notation. Each node is syntactically nested with all its attributes and its child nodes to ease configuration for large-scale network experiments and, by accurately modeling the network stack, to support *high-fidelity* distributed experimentation.

Run-time Controller is responsible for controlling live experiments. This could include tasks like starting a DoS attack or altering the data polling pattern based on observed states at the remote site. However, overuse of dynamic adjustment may increase communication overhead, lowering *performance*. Therefore, a good practice may be to shift the majority of cross-testbed communication to initialization and cleanup stages.

Error Detector is responsible for detecting abnormalities that occur in a live experiment. These could include errors such as host failure, asynchronization, or known warnings due to certain experimental parameters/intermediate outputs exceeding a preset threshold. To ensure *fidelity*, the system will then take corresponding actions such as relocating extra hosts, generating local or cross-testbed alerts, writing events to local system logs, or terminating/restarting experiments. The error detector can either be triggered by an abnormal event or perform periodic checks to ensure a healthy experimental environment.

Data Plane Configurator is responsible for issuing controls to the data plane at the initialization, run-time and cleanup stages of an experiment. Controls include setting the distribution of incoming traffic, how to aggregate outgoing traffic, and specifying the types of reports to collect upon completion of an experiment.

5.3.2. Model Data Plane

Traffic Distributor is responsible for bridging traffic data across two testbeds based on the settings from the data plane configurator. The number of cross-testbed physical links are minimized for *performance* gain. Therefore, traffic data is often aggregated and the traffic distributor is used to forward traffic to the right destination. VPST-C has one type of node called a "super node", which handles real traffic from the emulation channel. It appends a new virtual IP header based on an IP translation table established at the initialization stage by the resource allocator, and then distributes packets as if the super node is a traffic generator. Upon leaving VPST-C, the virtual IP header is extracted so that a remote real testbed can correctly handle the packet.

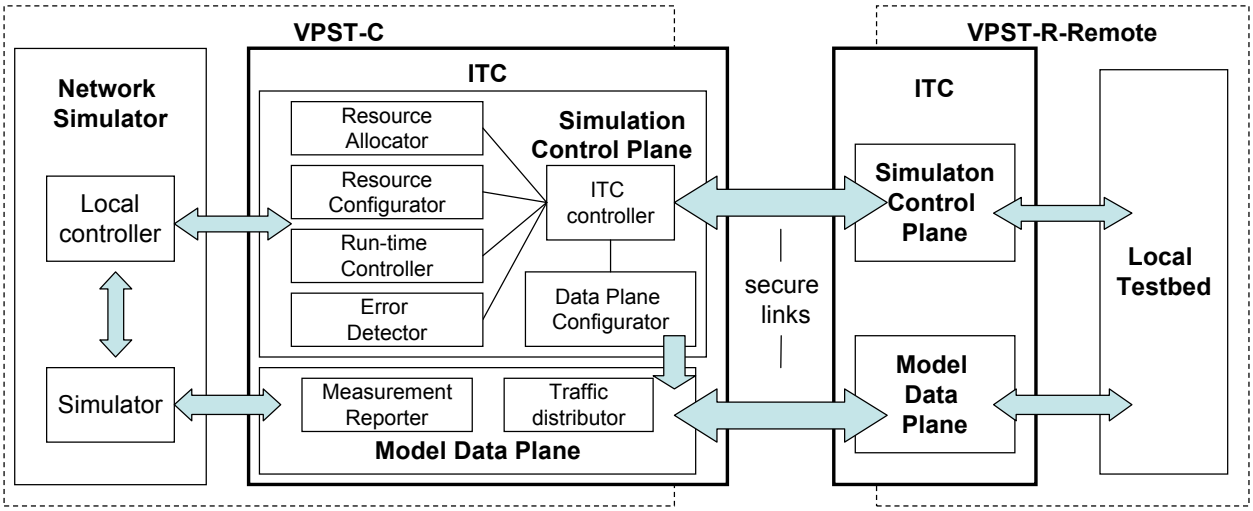
Measurement Reporter is responsible for collecting statistics of the experiment ranging from a single value like average packet loss rate to detailed per-host status reports. Upon completion of an experiment, reports are sent to the remote testbeds as instructed by the control plane. This module is included in the data plane because in some cases, remote experimenters may need the entire trace files, which require large bandwidth. Meanwhile, the security level of the data channel must be raised accordingly if the reports contain sensitive data.

6. Related Work

DETER[4] is an Emulab-based security testbed with a shared infrastructure of several hundred experimental nodes. Many attacking/malware models and tools for traffic/topology generation and analysis are available through DETER. These resources must be leveraged in order to faithfully test the security of SCADA networks. DETER also supports remote access while providing assurance for isolation and containment of each experiment. However, everything in DETER is real, from the operating system to the network stack, which makes replicating a SCADA network infeasible.

National SCADA Test Bed Program (NSTB)[2] was jointly established at INL and Sandia National Laboratory (SNL). NSTB consists of 61 miles of cables, 7 outstations, and more than 300 monitoring points across the nation. Many equipment manufacturers and government agencies conduct tests on NSTB for finding tangible solutions to growing threats to the power grid. However, NSTB is not publicly available. Even if it were, it is still insufficient to explore the impact of various security technologies on a nation-wide power grid. In addition, it lacks the flexibility to explore different architectures,

Figure 2. ITC Architecture Diagram



since it is primarily a physical system.

Virtual Control System Environment Project (VCSE) [8] in SNL was designed to incorporate their existing tools, including simulated, emulated, and physical components to assess security vulnerabilities in SCADA systems. The main difference from VPST is that its OPNET-based simulation framework does not scale nearly as well as RINSE.

7. Conclusion and Future Work

In this paper, we have shown that there is potential in connecting VPST with remote testbeds, in order to take advantage of their unique offerings. We have also discussed many of the concerns regarding integrating multiple testbeds as well as the methods that we employ to alleviate said concerns. One of the next steps we would like to take is to develop a black-box implementation of the current ITC. This box would be installed in a remote testbed and would be responsible for seamlessly connecting a remote testbed to ours with as little manual configuration as possible. In addition to this, we believe there is benefit in extracting as much efficiency out of the WAN transmissions as possible (e.g., compression of data and intelligent use of control messages to reduce the amount of traffic that must pass over the link). For instance, a real flooding attack would not be tolerated over a WAN link since this will likely be the bottleneck.

Acknowledgments

We thank Susan Hinrichs for many constructive suggestions. This work was supported in part by a grant from the National Science Foundation (CNS-0524695).

References

[1] Dnp3 specification, secure authentication, supplement to volume 2. *http://www.dnp.org/Modules/Library/Document.aspx*.

[2] National scada test bed program. *http://www.inl.gov/scada/publications/index.shtml*.

[3] Powerworld simulator. *http://www.powerworld.com/*.

[4] T. Benzel, R. Braden, D. Kim, C. Neuman, A. Joseph, K. Sklower, R. Ostrenga, and S. Schwab. Experience with deter: a testbed for security research. pages 10 pp.–388, 0-0 2006.

[5] DNP.org. Dnp: Distributed network protocol. *http://www.dnp.org*.

[6] W. Hwu, W. Sanders, R. Iyer, and K. Nahrstedt. Trusted illiac: A configurable, application-aware, high-performance platform for trustworthy computing. *http://www.iti.illinois.edu/sites/default/files/docs/cri-snowbird-06-talk-final.pdf*.

[7] M. Liljenstam, J. Liu, D. Nicol, Y. Yuan, G. Yan, and C. Grier. Rinse: The real-time immersive network simulation environment for network security exercises. In *PADS '05: Proceedings of the 19th Workshop on Principles of Advanced and Distributed Simulation*, pages 119–128, Washington, DC, USA, 2005. IEEE Computer Society.

[8] M. J. McDonald, G. N. Conrad, T. C. Service, and R. H. Cassidy. Cyber effects analysis using vcse. *Tech. Rep. SAND2008-5954, Sandia National Laboratories, September 2008*.

[9] D. M. Nicol, C. M. Davis, and T. Overbye. A virtual power system testbed for cyber-security decision support. *Proceedings of the 2009 INFORMS Simulation Society Workshop on Simulation: At the Interface of Modeling and Anaylsis*.

[10] OPSAID. Department of energy office of electric delivery and reliability's national scada testbed program. *Initial Design and Testing Report*.

[11] PNNL. Looking back at the august 2003 blackout. *http://eioc.pnl.gov/research/2003blackout.stm*.

[12] UIUC. Trustworthy cyber infrastructure for the power grid. *http://tcip.iti.illinois.edu*.