Risk Modelling the Transition of SCADA System to IPv6

Suriadi Suriadi, Alan Tickle, Ejaz Ahmed, Jason Smith, Hasmukh Morarji

Risk Modelling the Transition of SCADA System to IPv6

Suriadi Suriadi, Alan Tickle, Ejaz Ahmed, Jason Smith and Hasmukh Morarji

Queensland University of Technology, GPO Box 2434,
Brisbane, QLD 4001, Australia
s.suriadi@qut.edu.au, ab.tickle@qut.edu.au, e.ahmed@qut.edu.au,
j4.smith@qut.edu.au, h.morarji@qut.edu.au

Abstract. SCADA is one of a set of manufacturing-and-control systems that are used to monitor and control critical infrastructure. Such systems extensively utilise communications network protocols such as TCP/IP to interconnect a diverse array of components. A major forthcoming change within TCP/IP is the adoption of the IPv6 protocol and inevitably this change will affect SCADA systems. However IPv6 introduces its own set of vulnerabilities. Hence, given the scale and complexity of current SCADA systems, there is a need for organisations to be able to model and review the risks emanating from the propagation of identifiable vulnerabilities in IPv6 prior to actual operational deployment. This work shows how the required tools can be constructed by complementing the Information Security Management (ISM) risk modelling tool with the formal technique of Coloured Petri Nets (CPN). The results of the application of the tools in a case study confirm the utility of the approach.

Keywords: SCADA, IPv6, Risk Modelling, Formal Methods, Coloured Petri Nets.

1 Introduction

SCADA is one of a set of manufacturing and control systems used to monitor and control critical infrastructure particularly in utilities such as energy and water [1]. One of the key enabling technologies underpinning such systems is the TCP/IP suite of network protocols which facilitates the interconnection of a diverse array of components such as those found in manufacturing and control systems. Within the Internet community, the impending change from the current version of the IP protocol (IPv4) to the IPv6 implementation offers a number of potential benefits to corporate networks and the Internet at large [2, 3], including SCADA systems. For example, the significant increase in the number of public IP addresses and the auto address configuration feature of IPv6 make it possible to deploy hundreds, if not thousands, of SCADA components, such as smart meters, in a relatively efficient manner. Furthermore, the anycast addressing feature of IPv6 may also increase the reliability of a SCADA system.

However, any new technology comes with its own inherent vulnerabilities. For IPv6 this is true both in its ultimate deployment as well as during the transition from IPv4 [3]. Currently, the opportunities for organisations to examine the behaviour of their SCADA systems under IPv6 are limited to test systems [1] which makes it difficult to predict how the operational SCADA over IPv6 systems will behave.

This paper discusses some of the results of a project to apply a set of tools that could be used to perform a risk analysis of operational SCADA systems. There are two processes involved: (1) a study of how IPv6 vulnerabilities can manifest themselves as threats to a SCADA system running over IPv6, and (2) an analysis of the level of risks that the identified threats pose to the system.

The main contribution of this paper is the proposal of an approach to combine a well-known formal method of Coloured Petri Nets (CPN) [4] with a risk analysis tool, called the Information Security Management (ISM) Tool [5,13,15], in executing the two processes identified above. A formal method approach is used because the underlying mathematical nature of formal modelling and analysis enables a precise system modelling and potentially complete threat detection, depending on the validity and the details of the model of the system. Furthermore, it is also a cheaper and less time consuming in comparison to deploying a real SCADA system.

Once the threats of IPv6 vulnerabilities to a SCADA system are identified using CPN, we can then feed these threats into the ISM Tool for risk analysis. Knowing threats alone is not sufficient as, ultimately, the goal of threats detection and identification is to allow the management to allocate resources proportionately to the risk that these threats carry. Thus, a risk analysis is needed. The ISM Tool is chosen as a risk analysis tool mainly because of its flexibility: it can be used as threats documentation tool as well as a risk model generation tool which facilitates straightforward risk analysis. It also provides a simple graphical interface which makes it easy to use. Furthermore, it has a long and solid development history of over 15 years [15-17].

This paper starts with an overview of the vulnerabilities of IPv6 (Section 2) followed by an explanation of how we can detect potential threats to SCADA systems resulting from IPv6 vulnerabilities using the formal technique of CPN (Section 3). This is followed by a description of the ISM Tool for analysing risks of SCADA systems deployed over IPv6 networks (Section 4). The result of a case study of using the ISM Tool is detailed in Section 5, followed by a conclusion in Section 6.

2 Known IPv6 Vulnerabilities

While IPv6 offers a number of significant improvements over IPv4, it also introduces a number of new vulnerabilities [6]. These vulnerabilities can be broadly categorised into two main groups. The first group comprises vulnerabilities that are intrinsic to the IPv6 itself. The second group of IPv6 vulnerabilities arises from the IPv4 to IPv6 transition where the two protocols stacks must co-exist.

Of the vulnerabilities that are intrinsic to IPv6, there are three that potentially are of greatest concern in SCADA networks. The first of these vulnerabilities arises from the Neighbour Discovery (ND) and Address Auto-configuration mechanism (RFC

2461 [7] and RFC 3756 [8]). The second key vulnerability arises from the adoption of the Mobile IPv6 extensions which were an optional add-on feature for IPv4 networks but which are an implicit feature for IPv6 networks (RFC 3775[9]). The third important vulnerability arises from IPv6 extension headers. In addition to the vulnerabilities that are intrinsic to IPv6, there are also vulnerabilities that arise as a result of IPv4 and IPv6 co-existence (RFC 4942 [10]). For the purposes of illustration, in this paper we limit our discussion to the vulnerabilities arising from the IPv4 to IPv6 transition and the Neighbour Discovery (ND) mechanism.

Within IPv6, the role of the Neighbour Discovery (ND) mechanism is to create and maintain the mapping between the IP-layer address and the link-layer address. This is achieved through a process of Neighbour Solicitation (NS) and Neighbour Advertisement (NA). However this same process offers a vector through which an attacker is able to compromise system integrity by redirecting traffic away from the intended recipient. This attack could, for example, be initiated by injecting spoofed NA and/or NS messages either remotely or locally thereby corrupting the link between IP address and the corresponding link-layer address (see RFC 3756 and RFC 3964 for details [8, 11]). While such an attack is similar to the ARP spoofing attack within IPv4, there are few known mitigation techniques for the NA spoofing attack [12] particularly since, in IPv6, such an attack can be launched remotely (see RFC 3964 [11] for details).

An alternative vector of attack and one that could potentially arise in the IPv4-IPv6 transition process is to exploit the 6to4 tunnelling process. An attacker can spoof the source address on the inner IPv6 packet to a victim's address (for example, a 6to4 relay router). Without proper security checks, the attacker's IPv4 address (which is contained in the outer IPv4 packet) is discarded when the outer IPv4 header is de-capsulated. The net effect is (1) to make the attackers' actual IP address untraceable, and (2) to 'reflect' reply packets to the victim's IP address thereby creating a (distributed) denial of service attack as described in RFC 3964 [11]. As will be discussed, the key purpose of the work undertaken is to demonstrate how to detect threats and analyse their risks should the vulnerabilities described above be exploited in a control system environment.

3 Detection and Identification of IPv6 Threats to SCADA Systems

In Section 2, we described the two selected IPv6 vulnerabilities viz. IPv6 Neighbour Discovery (ND) vulnerabilities and IPv4-to-IPv6 transition mechanism vulnerabilities. However, in order to use the ISM risk modelling tool to perform a risk analysis of a SCADA system running on IPv6 network, the threats (as well as their propagations and consequences) resulting from these IPv6 vulnerabilities need to be clearly identified. Such an understanding is crucial because *risk* by definition is a function of both the probability and the consequences of identified threats to the system examined [13]. Therefore, the ability to detect and thus identify these threats is a precursor to risk analysis.

There are several techniques that can be used to study how IPv6 vulnerabilities can manifest themselves as threats to a SCADA system. These include:

1. use the presence of IPv6 in current COTS (commercial-off-the-shelf) network products to deploy a prototype SCADA over an working IPv6 system thereby allowing threats resulting from IPv6 vulnerabilities to be identified directly,
2. use formal analysis techniques to model a SCADA over IPv6 system and identify the vulnerabilities, and
3. use informal analysis of how IPv6 vulnerabilities can lead to threats to SCADA over IPv6 by studying the existing literature [6, 14] and drawing upon past experience.

Obviously, the first approach provides the most authoritative detection and identification of threats. However, at the moment the absence of an accessible real-world deployment of SCADA over IPv6 renders the option somewhat problematic. It would also be costly and time consuming.

On the other hand, the last approach (informal analysis) is comparatively cheap and not as time consuming. Provided there exists a detailed documentation of a SCADA system and IPv6 technologies (and their vulnerabilities), it is feasible to predict how a SCADA system will react when vulnerabilities are exploited. In fact, this is the approach that has been used in many of existing publications of security analysis of IPv6 such as RFC 3756 and RFC 3964 [8, 11]. However, while informal analysis is less costly and less time-consuming than using a 'live system', such an approach is not exhaustive. It also suffers from having too many variables to consider thereby rendering the approach incomplete and prone to errors.

The second approach (using formal methods) is arguably the best approach of the three techniques. It is comparatively cheaper and less time consuming than deploying a real SCADA system and the mathematical nature of formal modelling and analysis delivers a precise system modelling and potentially complete threat analysis depending on the validity and the details of the model of the system.

In this paper, we use both the formal analysis (Coloured Petri Nets (CPN) [4] along with the supported simulation and state space analysis) and informal analysis. In particular we show how CPN can be used to detect and identity threats manifested from IPv6 vulnerabilities and how the identified threats can then be documented and used as inputs to the ISM Tool for risk analysis. Alternatively, when a threat analysis has been performed informally, the results can be verified using the formal methods just mentioned in the future.

In the remainder of this section, we explain the result of our formal and informal analysis on how IPv6 vulnerabilities can manifest themselves as threats to a SCADA system over IPv6. We present the resulting threats from the three IPv6 vulnerabilities explained in Section 2: (1) NA/NS spoofing vulnerability (RFC 3756 [8]), (2) remote injection of ND messages vulnerability (RFC 3964 [11]) , and (3) reflection of traffic to 6to4 relay vulnerability (RFC 3964 [11]).

We have developed a proof-of-concept CPN model to detect threats resulting from the first vulnerability. The threats from the other two vulnerabilities have been informally analysed. From our analysis, we have managed to detect how the IPv6 vulnerabilities studied can become threats to SCADA over IPv6, and how those threats can propagate concluding in the SCADA system reaching a potentially undesirable state. The results of our formal and informal analysis are summarized in Table 1 and Table 2, and are fed into the ISM Tool for risk analysis (details are in the remainder of this paper).

3.1 Threats Resulting from NA/NS Spoofing Vulnerability

The immediate threat posed by NA/NS spoofing is that a legitimate node (such as a Master controller node – see Fig. 1) obtains an incorrect link layer address of its neighbours. This situation may lead to control messages being delivered to an incorrect RTU2 resulting in RTU2 executing the control messages intended for RTU1 and hence contributing to system instability and potential failure. .

A brief description of the proof-of-concept CPN model for NA spoofing vulnerability, and how the model can tell us the threats manifested from this vulnerability, is provided. A simple SCADA system is studied (see Fig. 1). The `Industrial Control Network' part of the overall SCADA (see top right corner of Fig. 1) has been modelled using CPN. This network consists of one master controller with two RTUs, along with other supporting equipment.

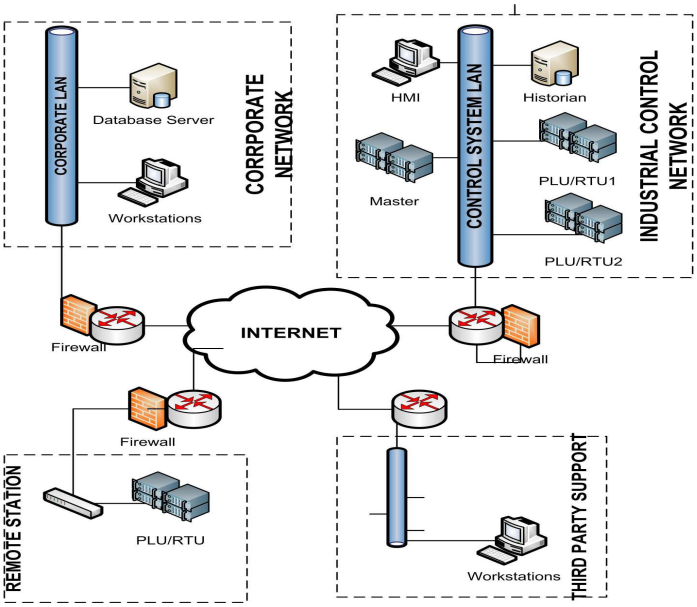

Fig. 1. A Simple SCADA system. The `Industrial Control Network' is modelled using CPN.

Fig. 2 shows a snippet of the industrial control network modelled in CPN. It depicts a subset of RTU2 operations. Firstly, we model the sending of a Neighbour Advertisement (NA) message (see the `Sending NA' transition inside a rectangle box on the top left corner of Fig. 2). We also model the receiving of an NA message (and the updating of the Neighbour Cache Table) at each node, and the operations that RTU2 execute when a control message is received (see the lower part of Fig. 2).

As this is a proof-of-concept model, we abstract the NA packet and the control message packet to filter out information that is not relevant for our purpose. Similarly, we also do not model how RTU2 exactly processes a received control message; rather, we are more interested in how RTU2 can reach an unstable/undesirable state.

In the model, we assume that RTU2 has almost reached its `maximum capacity' (in a real world system, imagine RTU2 as a controller of an actuator, such as a pump, that pumps oil into an almost-full tank). Using the above analogy, we model RTU2 such that it can only handle additional 3 `pump' control messages before the tank is full.

Overall, we model the system such that RTU2 will send an NA message to be processed by both RTU1 and Master controller. After a successful processing of the NA messages by all nodes, the Master controller sends 5 control messages to RTU1.

Through the simulation of the CPN model, we have verified that the modelled SCADA system behaves correctly in the generation and processing of both NA and control messages.

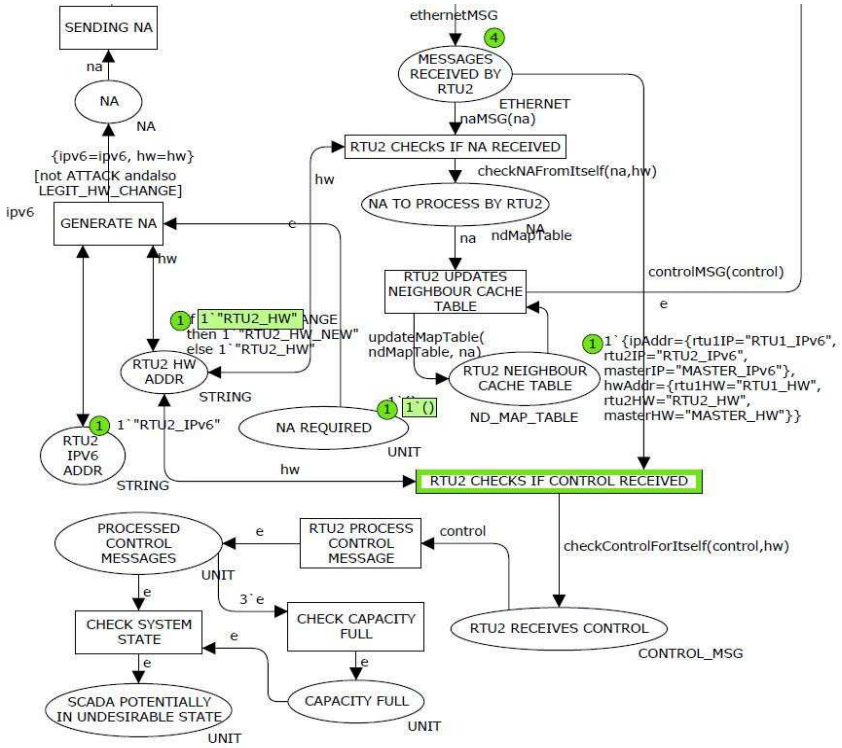

Fig. 2. A CPN Model of select RTU2 operations.

The next step is to model the exploitation of the spoofed NA vulnerability. We do so by adding an attacker model as shown in Fig. 3. We model the attacker to inject a spoofed NA message claiming that the corresponding hardware address of an IPv6 address (which actually belongs to RTU1) is the attacker's hardware address. Then, we model the attacker modifying any control messages it receives so that they are directed to the low-capacity RTU2 with the aim of causing instability in the system controlled by RTU2.

Leaving the rest of the model intact, we then run a simulation to study what happens to the SCADA system when the attacker injects the spoofed NA message followed by the Master controller sending a series of 5 control messages to RTU1. From the simulation of the model, we note that the control message is mistakenly delivered to RTU2 who then executes the control messages. As a result, the SCADA system may reach a potentially undesirable state (see Table 1 and 2 for summary).

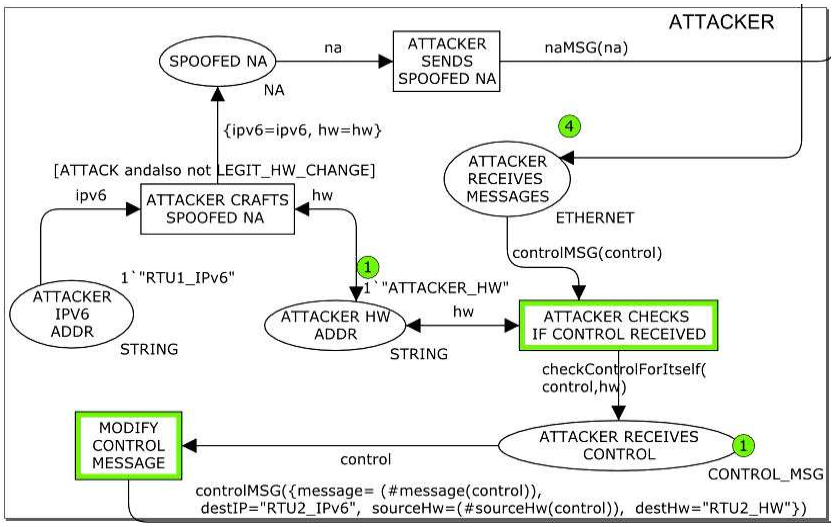

Fig. 3. A CPN Model Depicting an Attacker Exploiting IPv6 Spoofed NA Vulnerability.

3.2 Threats Resulting from Remote Injection of ND Messages Vulnerability

The use of tunnelling mechanism could allow a compromised external node (such as a third-party support workstation) to inject a spoofed RA (or NA) messages. This may result in a successful exploitation of ND vulnerabilities, such as the modification of default router information. The consequence of such an attack is that control messages sent to a remote station RTU can be modified by an attacker, resulting in incorrect control messages received by the remote RTU. This may result to an unstable SCADA system at the remote location which could lead to the system being in an undesirable state (see Table 1 and 2).

3.3 Threats Resulting from Reflection of Traffic to 6to4 Relay Vulnerability

The use of tunnelling mechanism could allow a compromised external node (such as a third-party support workstation) to reflect traffic to a 6to4 relay router (such as the one at the Industrial Network). This may result in the router being overwhelmed with

too many packets causing legitimate control messages to be dropped, leading to instability and an undesirable state (see Table 1 and 2).

4 The ISM Risk Modelling Tool

In Section 3, we have described how we can identify threats to a SCADA system based on known IPv6 vulnerabilities. However, at least from a management point of view, the ultimate goal from this exercise is to be able to proportionately allocate resources to mitigate the identified threats according to their risks. To this end, we need a mechanism to study how 'risky' the identified threats are.

The ISM Risk Modelling tool [13, 15] has been developed for this purpose. This tool is developed based on the *information security management and modelling* approach proposed by Longley et al and has been developed over more than 15 years [15-17]. This tool can be used for several purposes, including: (1) as a tool to aid threat documentation, (2) as a risk modelling tool: by reading the threat documentation that has been fed into the ISM Tool, and (3) as a tool to analyse the effectiveness of threat countermeasures.

Threat Documentation: Depending on the system being studied, organizations may already have such threat documentation, or, threats can be identified and documented using the approach explained in Section 3. The ISM Tool allows these threats to be properly documented in a form that can be readily used for risk analysis. In Section 5, we provide an example of how threat documentation and threat propagation are documented using ISM Tool.

Risk Simulation and Analysis: Once threats are documented into the ISM Tool, they can be automatically scanned to generate a risk simulation by creating a threat network diagram [15] which shows both the risk measurement of a threat and the propagation of threat. The risk measurement is derived based on the consequences of a threat documented (catastrophic, major, moderate, minor, or insignificant) as well as the probability of the threat happening [13]. In Section 5, we demonstrate how we can derive a risk simulation on a SCADA system over IPv6 network using the ISM Tool.

Threat Countermeasures Analysis: The ISM tool also captures the set of mitigation techniques that have been (or planned to be) applied. Similar to threat documentation, such information may already be well-documented in an organization, or, a further study (using methods described in Section 3) is needed to verify the effectiveness of a countermeasure technique. An analysis of the effectiveness of countermeasure techniques to mitigate IPv6 threats to SCADA systems is part of the future work.

5 A Case Study Using the ISM Tool

In this section, we demonstrate how we can use the ISM Tool to (1) document detected threats and threats propagation, and (2) generate a simulation of risk which shows the risk level of documented threats.

The risk simulation can be performed in two styles: 'forward' and 'backward'. In the 'forward' style, a user chooses the starting threats that are assumed to have happened, and then either (a) uses the tool to show the ensuing chains of threats (as well as the risk level for the ensuing threats), or, (b) chooses the causal threats so that ISM Tool can show how the starting threats may lead to the chosen causal threats as well as the risk level associated with the starting threats, the propagated threats, and the causal threats. In the 'backward' style, a user simply chooses the causal threat. From there, the ISM Tool will do a 'backward' analysis to show the threats that may lead to the chosen causal threat [15], their propagation, and their risk levels.

5.1 Threat Documentation Using ISM Tool

To document threats, some key information is required, including (1) the entities of the system (such as platforms, hardware, softwares, networks, information assets, and so on) which may become the victims of some threats, (2) the relationships between those entities (for example, a master controller and an RTU may have a communication relationship), and (3) the detected threats and their propagations.

The first two pieces of information (entities involved and their relationships) can be easily documented based on the simple SCADA system shown in Fig. 1. Furthermore, information about the detected threats and their propagations is obtained from the results of both the formal and informal analysis of IPv6 threat detection explained in Section 3. These results are summarized in Table 1 and Table 2.

5.2 Risk Simulation and Analysis Using ISM Tool

Once the threats are documented, it is simple to obtain the risk simulation. Fig. 4 shows a threat network generated using a `forward' risk simulation whereby we determine the set of initiating threats ('Incorrect Link Address', 'Packets Reflected to IPv6 Relay Router', and 'Local Nodes Subjected to Remote ND Attacks') and use the ISM Tool to identify how the SCADA system can reach an undesirable state. The colour of each of the node in Fig. 4 is used to convey the risk level of each of the threat (green colour denotes the initiating threats, grey colour denotes a low risk level, yellow colour denotes a high risk level, and red colour denotes an extreme risk level). The ISM Tool can also generate a risk measurement report (see Table 3) which summarizes the resulting threats (including the initiative threats, the intermediary threats, and the resulting threats) and their risk level. From Fig. 4 and Table 3, we can conclude that there is an extreme risk of a seemingly-minor threat (e.g. a node obtains incorrect link layer address) leading to a major threat whereby a SCADA system becomes unstable. A `backward' style analysis can also be performed to determine the threats that could cause the SCADA system reach an undesirable state.

Table 1. A Summary of Detected Threat Entities

Threat Name	Threat Entity	Consequences
Incorrect Link Address	• RTU/Master/HMI/Historian at Industrial Control Network (ICN) • RTU at Remote Station	Minor
Internal Control Messages Communication Errors	• Communication between Master Controller and RTU1/RTU2 at ICN	Moderate
HMI Provides Inaccurate Feedback	• HMI at ICN	Moderate
Incorrect Control Message Sent	• Master Controller at ICN	Moderate
Packets Reflected to IPv6 Relay Router	• Router/Firewall at ICN	Moderate
IPv6 Hosts Receive Too Many Illegitimate Packets	• Industrial Control Network Remote Station	Moderate
Valid Control Messages Dropped	• Industrial Control Network • Remote Station	Moderate
Local Nodes Subjected to Remote ND Attacks	• RTU1/RTU2/Master at ICN	Moderate

Table 2. A Summary Threat Propagation

Incident Threat Entity	Target Threat Entity
Incorrect Link Address	Internal Messages Communication Errors
Incorrect Link Address	Internet Messages Communication Errors
Internal Messages Communication Errors	HMI Provides Inaccurate Feedback
Internet Messages Communication Errors	HMI Provides Inaccurate Feedback
HMI Provides Inaccurate Feedback	Incorrect Control Message Sent
Poorly Secured Host	Compromised Host
Compromised Host	Incorrect Control Message Sent
Compromised Host	Packets Reflected to IPv6 Relay Router
Packets Reflected to IPv6 Relay Router	Valid Control Messages Dropped
Valid Control Messages Dropped	HMI Provides Inaccurate Feedback
Local Nodes Subjected to Remote ND Attacks	Incorrect Control Message Sent
Incorrect Control Message Sent	SCADA Reaches Undesirable State

Table 3. An Example of Risk Measurement Summary for Fig. 4

Threat	Risk
Incorrect Link Address	High
Internal Control Messages Communication Errors	Extreme
Internet Control Messages Communication Errors	Extreme
Local Nodes Subjected to Remote ND Attacks	Extreme
Packets Reflected to IPv6 Relay Router	Extreme
HMI Provides Inaccurate Feedback	Extreme
Valid Control Messages Dropped	Extreme
Incorrect Control Message Sent	Extreme
SCADA Reaches Undesirable State	Extreme

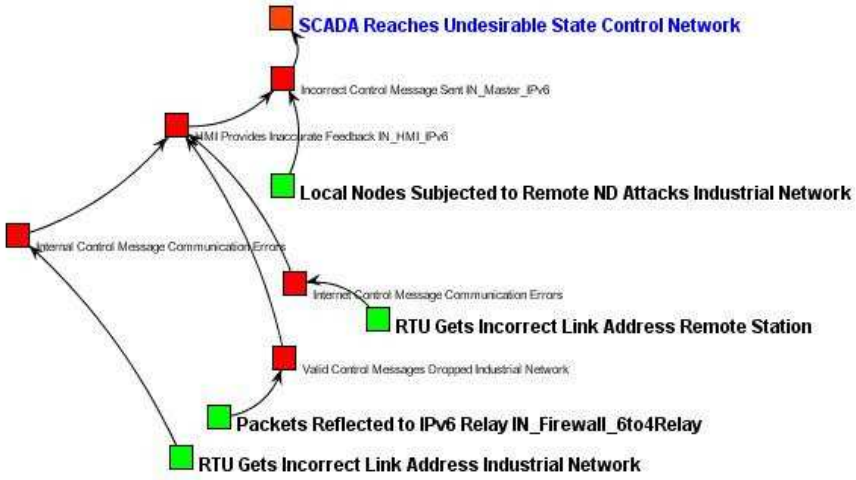

Fig. 4. Risk Simulation using the ISM Tool – 'forward' risk simulation.

6 Conclusion

We have shown tools and techniques that can be used to perform a risk analysis of a SCADA system running on IPv6 network. In particular, we have shown how a formal technique, such as Coloured Petri Net, can complement the use of ISM Tool: as a technique to detect and identify threats manifested from IPv6 vulnerabilities which are subsequently fed to the ISM tool for risk analysis. We have also shown how the ISM Tool can be used to simulate and analyse risks in a SCADA system. An interesting future work will be to develop a detailed CPN model of a representative SCADA system running on IPv6 to allow threats manifested from IPv6 vulnerabilities to be comprehensively detected and documented.

Acknowledgment. This work was supported by the Australia-India Strategic Research Fund 2008-2011.

References

1. Wiles, J., Drake, P., and Claypoole, T.: Techno Security's Guide to Securing SCADA: A Comprehensive Handbook On Protecting The Critical Infrastructure. Syngress, Rockland, Mass (2007).
2. United States Government Accountability Office, Internet Protocol Version 6: Federal Agencies Need to Plan for Transition and Manage Security Risks (2005).
3. United States Department of Commerce, The Evolving Internet: A Technical and Economic Assessment of Internet Protocol, Version 6 (IPv6) (2006).

4. Jensen, K.: A Brief Introduction to Coloured Petri Nets. In: E. Brinksma (ed.) Proceedings of the Third international Workshop on Tools and Algorithms For Construction and Analysis of Systems (April 02 - 04, 1997). Lecture Notes in Computer Science, vol. 1217, pp. 203-208. Springer-Verlag, London (1997).
5. Branagan, M., Dawson, R., Longley, D.: User Documentation - ISM (Information Security Management) Risk Management Tool (2006).
6. Leeuwen, B.V.: Impacts of IPv6 on Infrastructure Control Systems. Sandia Report SAND2007-0383P. Sandia National Laboratories (2007), http://www.sandia.gov/scada/documents/VanLeeuwen-2007-0383P.pdf.
7. Narten, T., Nordmark, E., Simpson, W., Neighbor: Discovery for IPVersion 6 (IPv6), RFC 2461. The Internet Society (1998), http://www.ietf.org/rfc/rfc2461.txt.
8. Nikander, P., Kempf, J., Nordmark, E.: IPv6 Neighbor Discovery (ND) Trust Models and Threats, RFC 3756. The Internet Society (2004), http://www.ietf.org/rfc/rfc2461.txt.
9. Johnson, D., Perkins, C., Arkko, J.: Mobility Support in IPv6. RFC 3775. The Internet Society (2004), http://www.ietf.org/rfc/rfc3775.txt.
10. Davies, E., Krishnan, S., Savola, P.: IPv6 Transition/Co-existence Security Considerations, RFC 4942. The Internet Society (2007), http://www.ietf.org/rfc/rfc4942.txt.
11. Savola, P., Patel, C.: Security Considerations for 6to4, RFC 3964 (Informational) (2004).
12. Convery, S., Miller, D.: IPv6 and IPv4 Threat Comparison and Best-Practice Evaluation. Cisco Systems White Paper (2004).
13. Dawson, R.E.: Secure communications for critical infrastructure control systems. Queensland University of Technology, Brisbane, QLD (2008).
14. Stouffer, K., Falco, J., Kent, K.: Guide to Supervisory Control and Data Acquisition (SCADA) and Industrial Control Systems Security. In: Recommendations of the National Institute of Standards and Technology (2006).
15. Branagan, M., Dawson, R., Longley, D.: Security Risk Analysis for Complex Systems. In: Proceedings of the Information Security for South Africa 2006 from Insight to Foresight Conference. ISSA, Pretoria, South Africa (2006).
16. Caelli, W.J., Longley, D., Tickle, A.B.: A Methodology for Describing Information and Physical Security Architectures. In: Eight International Conference on Information Security (SEC) 1992. IFIP Transactions, Singapore (1992).
17. Kwok, L.F., Longley, D.: Information security management and modeling. Information Management and Computer Security, 7(1), 30-39 (1999).