T-ABAC: An Attribute-Based Access Control Model for Real-Time Availability in Highly Dynamic Systems

Mike Burmester
Department of Computer Science
Florida State University
Tallahassee, Florida 32306–4530
Email: burmester@cs.fsu.edu

Emmanouil Magkos and Vassilis Chrissikopoulos
Department of Informatics
Ionian University
Platia Tsirigoti 7, 49100 Corfu, Greece
Email:{emagos,vchris}@ionio.gr

***Abstract*—In highly dynamic systems resources may have to be accessed in real-time, within the strict time limits of underlying physical processes, with availability becoming critical. Current access control models such as RBAC and ABAC do not address real-time availability in a scalable way for such scenarios. In this paper we propose a real-time attribute-based access control model that extends the functionality of ABAC by using real-time attributes that reflect the requirements of critical applications. We describe two applications of our model: (a) a substation automation system, and (b) a medical cyber-physical system.**

***Keywords*- Dynamic systems, access control, real-time availability, cyber-physical systems, trusted computing.**

I. Introduction

Nowadays, most critical processes are supported by pervasive devices (*e.g.*, RFID tags, sensors, actuators), networked with each other and with other entities through a variety of network technologies and standard communication protocols. For example in cyber-physical systems (CPS) such as smart grids [1] and power plants [2] a bridge between the physical and cyber world is established making it possible for cyber systems to monitor and control physical devices. This involves measuring physical context data (location, temperature etc) and physical dynamics (power consumption, physiological data etc). In particular CPS deal with real-time events, either system-based or Nature-based, which cannot always be predicted or emulated in a scalable way. Sharing information in real-time is particularly challenging because it involves managing trust associations that have to be adjusted to take into account highly dynamic events.

The distribution of a critical process to a variety of smart objects entails several risks related to both system and networking aspects of a CPS [3], [4], [5]. In particular, the confidentiality, integrity and availability of communicated or stored data is often the target of local and/or remote adversaries, but is also impacted by random or benign faults and failures of cyber-physical components [6]. Traditional computer and network security fails to address in a unified manner how systems can outlive (survivability) unintentional (*e.g.*, human errors) or malicious unpredictable events, but also unexpected Nature-based events in real-time.

Traditional access control models [7], [8], [9], [10], [11], [12], [13] primarily focus on confidentiality and integrity. These models do not address scenarios in which resources are not available in time, *e.g.,* a few milliseconds too late. In particular, scenarios where sudden events that threaten the state of a system need to be addressed within strict time limits imposed by physical processes. Typically they assume either a static policy framework, or are restricted to a relatively small set of possible events, that is not appropriate for highly dynamic real-time applications.

Quality of Service (QoS) architectures [19], [20] extend Best Effort services for end-users of IP networks in terms of delay, jitter and loss. In highly dynamic systems an end-user may have to rely on a service response time $<4\,ms$ to provide its own response in $<10\,ms$—the time thresholds for IEC/TR 61850-90-5 compliant systems [22] (discussed in Section IV). For critical infrastructures, failure to address system malfunction in real-time may lead to catastrophic failure. In this case Best Effort is not sufficient, and a different approach is needed that takes into account the strict time constraints of physical processes.

Our contribution. We propose a real-Time Attribute-Based Access Control model (T-ABAC) that extends attribute-based access control by using real-time attributes that take into account the priority of access requests to support real-time availability within the strict time constraints of physical processes, against an active adversary. We show how our model can be used in cyber-physical applications that are protected by a Trusted Computing architecture to guarantee real-time availability for high priority IP packets, while also supporting integrity and confidentiality policies. We describe two possible applications of T-ABAC: (a) a substation automation system, and (b) a medical CPS.

The rest of this paper is as follows. In Section II we review access control models and Best Effort QoS architectures. In Section III we describe real-time attribute-based access control (T-ABAC). In Sections IV,V we present applications of T-ABAC for CPS systems, and in Section VI we conclude.

II. Related work

A. Access control models

Access control systems are trust infrastructures that manage resources of information systems. For example, the Bell-LaPadula model is a mandatory access control (MAC) model [7] that enforces need-to-know policies (confidentiality). Extensions of this model include the Chinese Wall [14] model that enforces separation of duties and the Biba model [8] that enforces integrity policies. Discretionary access control (DAC) models [9] support policies that allow the owner of an object to exercise control over that object. MAC and DAC models are not dynamic and are only used in coarsely grained security scenarios involving relatively small groups of subjects (users).

In role-based access control (RBAC) models, access permissions are assigned to roles, and roles to subjects [13]. A subject can exercise an access permission only if the permission is authorized for the activated role in a given session. RBAC addresses the requirements of multi-user and multi-application systems in large organizations, and simplifies administration of access rights. While RBAC scales better, it still is not suitable for highly dynamic applications, where unique roles have to be created for all combinations of security labels and constraints.[1] RBAC extensions [15], [16] address some temporal issues, but still do not scale well and cannot easily capture real-time security relevant information from the environment.

The attribute-based access control (ABAC) model [12], [17], [18] assigns attributes to subjects and objects. Authorization is defined for subject descriptors, consisting of several attribute conditions—see Figure 1. Permissions consist of a combination of an object descriptor that contains a set of attributes and attribute conditions, and an operation on the object(s) specified by the descriptor. Environment attributes such as, the time of day, temperature, etc, can also be used to control access. In its basic form, access to an object o by a subject s in a particular environment e is resolved by evaluating policy rules where each rule can be seen as a boolean function f evaluated over the attribute values of s, o, e contained in their descriptors: $access(s, o, e) \leftarrow f(attr(s), attr(o), attr(e))$. ABAC can encompass the functionality of RBAC by treating identities or roles as attributes [12].

Compared to RBAC, it captures dynamic environment attributes that are temporal. It can also enforce fine grained policies, specifically when the number of subject and object attributes becomes large. However in highly dynamic real-time applications the event space that determines the attribute values can be very large thus making any attempt to capture real-time availability scenarios non-scalable.

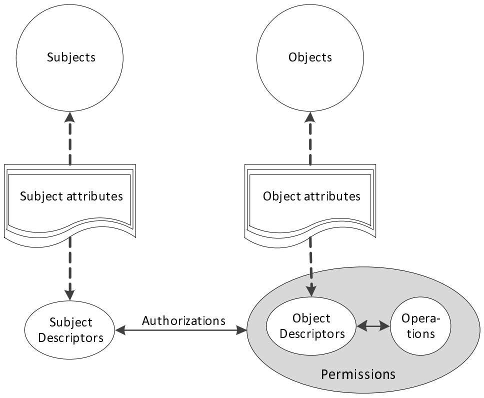

Figure 1. An overview of the ABAC model

B. Best Effort availability for network applications

The Internet Engineering Task Force (IETF) has proposed two Quality-of-Service (QoS) architectures that support Best Effort availability for Internet communication:

– DiffServ (Differentiated Services) [19], and
– IntServ (Integrated Services) [20].

DiffServ addresses per-hop forwarding behavior (PHB) of IP packets while IntServ addresses per-flow behavior of end-to-end streamed data. These architectures distinguish between core routers that queue and schedule packets, and edge routers that monitor and police traffic flows.

The header of a DiffServ IP packet is marked with a codepoint according to a service level agreement (SLA) between core and edge routers. This allows different bandwidths to be allocated to traffic on each link. For real-time service, core and edge routers must be trusted to adhere to the SLA, and core routers must be able to verify that packet markings are authentic. Since DiffServ does not provide any security for verifying the origin of packets or the authenticity of markings, it is vulnerable to bandwidth theft and illegal PHB promotion, and hence to denial of service (DoS).

IntServ uses a resource reservation protocol (RSVP) [21] to reserve flow state specification across a mesh of delivery paths that link the source to the destination(s). Nodes send messages with service availability spec requests downstream along uni-/multicast routes at regular intervals which spread through the network, and edge routers send corresponding RSVP messages upstream (or a reject message if they cannot support the requested reservation). Routers store this information (for short periods) and police flows. There are two parts in flow state specs: request specs that specify the guarantees needed, and traffic specs that contain the parameters of a token bucket algorithm. For real-time availability, core and edge routers must be trusted to adhere to committed reservations, and core routers must be able to verify that flows are authentic. IETF proposes authentication mechanisms for hop-by-hop integrity and node authentication. This will protect against corruption and spoofing of RSVP packets, but not against distributed DoS attacks.

[1] Our applications involve access to multicast network resources: for these the number of end-users groups is exponential in the number of end-users.

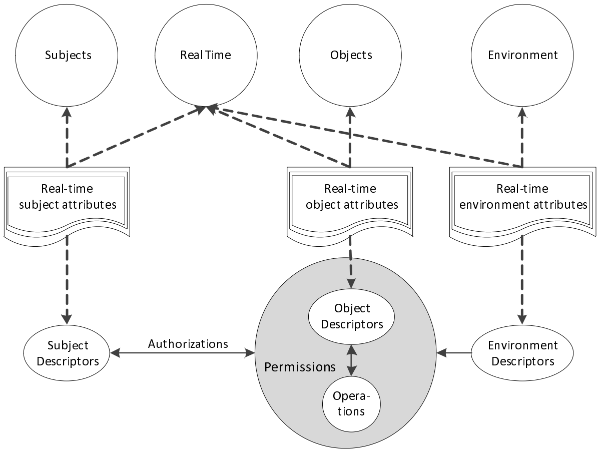

Figure 2. Overview of the T-ABAC model

In the following section we consider a network application that shares several features of DiffServ and IntServ, however focusing on guaranteed availability in the presence of a malicious adversary, within the real-time constraints of an underlying physical process. For such applications Best Effort solutions are not sufficient.

III. T-ABAC: Adding Real-Time to ABAC

There are several ways in which real-time can be added to an attribute-based access control system—Figure 2. In this paper we model real-time by an unbounded monotonic sequence of real numbers: $T = t_1, t_2, \ldots$, with $t_{i+1} - t_i = \delta > 0$, the time unit or latency. For highly dynamic systems such as the substation automation systems of an electricity grid, the latency δ is only a few milliseconds [22], while for smart environment systems δ can be several orders larger.

A real-time attribute is an attribute whose values depend on time $t_i \in T$. In our model, availability of a resource o at time t_i in environment e for a subject s is determined based on the real-time attributes $attr(x, t_i)$, $x \in \{s, o, e\}$, with values in a linearly ordered set of availability labels,

$$\mathcal{L} = \{\ell_m = hi \succeq \ell_{m-1} \succeq \ldots \succeq \ell_1 = lo\}.^{2}$$

The availability label of $attr(x, t_i)$ is called *priority* when x is a subject, *congestion* when x is an object, and *criticality* when x is the environment. Availability labels are dynamically determined based on user events, the context of the requested service and system events.

A. Need-to-get-now policies for TC-compliant networks

We focus on network applications, in particular IP mulicast networks. For these the resource o is a multicast IP packet P. We use a Trusted Computing (TC) architecture [9], [23], [24], and real-time attributes $attr(x, t_i)$ to determine availability for packets P transmitted at time $t_i \in T$. The priority of P is based on user events and the context of the packet P. The congestion is determined by the network traffic, and the criticality by environment events—*e.g.*, events that may cause system failure.

The threat model for such applications assumes an active adversary, with faults caused by Nature and/or the adversary. Since the network infrastructure is trusted, the adversary is restricted to DoS threats, which in this case must involve physical damage (communication is restricted to trusted parties/components). For robustness we assume that the network infrastructure (which includes core/edge routers) has sufficient redundancy to address such threats. This issue will be discussed further in our applications.

B. Real-time availability for TC-compliant networks

We adapt the DiffServ architecture to capture real-time availability. To control the per-hop forwarding behavior of edge routers, we assign to each packet a priority, to each edge router a congestion, and to each event a criticality. The following protocol enforces *need-to-get-now* policies for hi priority packets while supporting Best Effort QoS for other packets:

Packet forwarding protocol

Notation. R is an edge router; $P, P', P'', \ldots,$ are packets; $\rho^{in}, \rho^{in}_{max}$ are the rate and max rate of packets arriving at R; $\rho^{in}(hi)$ is the rate of hi priority packets arriving at R; ρ^{out}, ρ^{out}_{max} are the rate and max rate of packets forwarded by R; Q is the queue of R, a list $(P', P'', \ldots, P^{(k)})$ of $k \leq \rho^{out}_{max}$ packets, with $pri(P') \prec pri(P'') \prec \cdots \prec pri(P^{(k)})$.

Assumptions

Routing. Packets are forwarded using a routing protocol listed in the system specifications.

Available bandwidth. There is sufficient bandwidth to guarantee that: $\rho^{in}(hi) \leq \rho^{out}_{max}$.

Operations

Load P in Q: $load\,Q(P)$.

```
If $k < \rho^{out}_{max}$ then put $P$ in $Q$ so that it is the first packet in $Q$
        with its priority;
else if $pri(P') \prec pri(P)$ then drop $P'$ and put $P$ in $Q$ so
        that it is the first packet with its priority;
    else drop $P$.
```

Unload $P^{(k)}$ from Q: $unload\,Q$.
Remove lead packet $P^{(k)}$ from Q.

Protocol

```
while $\rho^{in} > 0$: $load\,Q(P)$, $P$ the next received packet.
while $Q \neq \lambda$: if $congestion \preceq pri(P^{(k)})$ then $unload\,Q$
        and forward packet $P^{(k)}$.
if $k < \rho^{out}_{max}$ or $pri(P^{(k)}) \prec congestion$ then lower by one
        the congestion level of $R$.
else increase by one the congestion level of $R$.
```

In this protocol when an edge router is congested, the need-to-get-now service overules Best Effort services (pos-

[2] For a more general model we may take $\mathcal{L}$ to be a lattice.

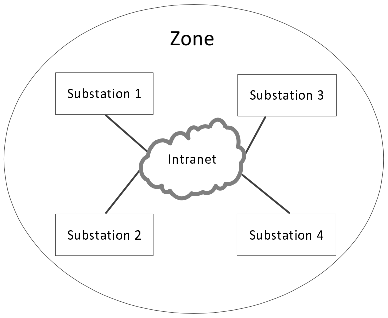

Figure 3. Inter-substation communication

sibly resulting in low priority packets getting dropped). This guarantees delivery of hi priority packets in real-time when there are no faults, provided that: (a) the specified rate of hi priority packets is not exceeded ($\rho^{in}(hi) \leq \rho^{out}_{max}$), (b) the network has sufficient redundancy to guarantee delivery of hi priority packets (up to the rate ρ^{out}_{max}). The guarantee extends to the case when there are faults (caused by Nature or the adversary), since for TC network architectures faults are reduced to DoS which is addressed by our redundancy requirements.

IV. Real-time availability for Substation Automation

Electricity grids are serviced by regions of interconnected transmission systems, with each region having several zones. The transmission of bulk power within a zone is controlled by substation automation systems (SAS) and substation-to-substation systems—Figure 3. The Technical Report IEC/TR 61850-90-5 [22] specifies the operational requirements for substation communication. For data exchange the IP transport protocol is used, with data encapsulated in IP packets. To allow for reporting information to be distributed in real-time to wide area network environments, packets are transmitted at regular intervals, typically every $4\,ms$.

The inter-operability of substations in real-time is essential for addressing faults of physical processes that may lead to system instability and blackouts. In particular, a packet reporting system failure must be received on time. For trusted inter-operability, we combine a real-time access control system with a Trusted Computing (TC) architecture based on the Trusted Platform Module (TPM) [23] and the Trusted Network Connect [24].

The TPM employs three trusted engines: *remote attestation*, *binding* and *sealing*, to assure the integrity of the system and to protect cryptographic tools and keys via *roots of trust*. The Trusted Network Connect requires the configuration of the OS of the sender/receiver (and the associated configuration data) to be checked prior to a communication channel being established. This guarantees that if the sender/receiver is compromised then no action is taken. A TPM architecture for SAS will therefore provide assurance for the integrity (and if required, the confidentiality) of the communication. However it will not provide assurance for real-time availability (in $4\,ms$).

A. *Real-time availability*

We capture real-time availability with a T-ABAC infrastructure. Since the SASs are TC-compliant, their network infrastructure is trusted. Then the packet forwarding protocol in Section III-B will guarantee availability in real-time for hi priority packets provided the specified rate is not exceeded and the network infrastructure has sufficient redundancy to guarantee delivery of hi priority packets, when there are no malicious faults.

IP packets are forwarded along a path of a mesh of delivery paths that link the source to the destination, as in the IntServ architecture (Section II-B), only in this case the routers store the routing information until it is updated. This is because the SAS network is not dynamic: changes essentially only involve service updates.

The priorities of IP packets are static, defined by their context/profile, whereas the congestion of router links is dynamic. For example, a hi priority packet has a payload that contains a message authentication code,[3] while congestion of the links is defined by the prevailing traffic flow.

B. *Requirements for real-time availability*

Dependability concerns real-time availability and integrity (and for some applications, confidentiality) for threat models with malicious adversaries. For TC-compliant systems, integrity is guaranteed by the TPM engines (remote attestation, binding and sealing) and by the fact that the network infrastructure is trusted. So for real-time dependability we only need to prove real-time availability which, for our application is restricted to hi priority IP packets. For this purpose we assume that the network infrastructure $\mathcal{N}$ of a SAS is designed to guarantee availability when only hi priority (authorized) packets are sent, and when there are no malicious faults. This requires that:

- the flow of hi priority packets via each router edge of $\mathcal{N}$ is bounded, and
- $\mathcal{N}$ has sufficient redundancy to cope with such traffic.[4]

External DoS attacks are addressed by our assumption that SASs are TC-compliant and the network $\mathcal{N}$ is trusted.

Consequently real-time availability is only impacted when authorized network traffic is congested (which could result from DoS attacks). When this happens the congestion label is raised: this favors transmission of hi priority (authorized) packets. Note that the adversary cannot inject hi priority packets because of the integrity checks (and a compromised TPM-compliant device is prevented from sending bogus hi priority packets). Using our earlier reliability assumption,

[3] The payload must include the priority label to prevent DoS attacks.

[4] To reduce the failure rate for availability, one may use error correction mechanisms. Message authentication codes are used to check integrity.

this will guarantee real-time availability. The level of protection afforded by this approach depends on:

1) the TPM interface (and in particular how well it is implemented), and
2) the additional redundancy in the SAS system design that is provided to address component failure due to natural or malicious events.

The last requirement can be quantified by: (a) assuming that the adversary cannot compromise or damage more than a certain number of system components at any point in time and, (b) specifying the natural events space $\mathcal{E}$ against which the system needs to be protected. Then we can reduce the likelihood of system failure by designing the system such that $1 - \Pr[\mathcal{E}]$ is sufficiently small. An acceptable failure rate for the electricity grid is $< 2^{-30}$.

The real-time availability of IEC61850-90-5-compliant systems was evaluated in [25] using a SISCO profiler [26] extended to capture security. A testbed with 17 TPM-compliant workstations running Ubuntu Linux and Windows was used. The average time required for AES encryption/decryption over 500 transmissions was $0.02294\,ms$ with standard deviation $\sigma = 0.00435\,ms$. For generating an HMAC it was $0.03425\,ms$ with $\sigma = 0.00172\,ms$. This bounds the end-to-end time $enc + dec + 2mac$ by $1.3\,ms$, which is $< 4\,ms$, the latency specified by IEC 61850-90-5. This shows realtime availability.

V. Real-time availability for a medical Cyber-Physical System

In a typical scenario of a medical cyber-physical system (MCPS) [27], medical devices act as sensors and actuators for continuously monitoring and controlling a patient's physiology. For example, a caregiver in a MCPS is able to control both the monitoring devices (*e.g.*, heart/respiratory rate, blood pressure) that passively monitor patient's vital signs and the delivery devices (*e.g.*, infusion pumps for pain control, ventilators, pacemakers) which may actuate treatment for changing the patient's state. Typically, a decision support entity will process the collected data and generate a smart alarm to alert clinicians when a vital sign crosses a predefined threshold. Alternatively, a decision support entity will utilize a smart controller to analyze the collected data and automatically initiate treatment (*e.g.*, drug infusion).

In large systems (*e.g.*, hospitals) raw medical information from multiple devices will be streamed into central locations to be processed in real-time. Smart alarm systems are also expected to go beyond the current threshold-based methods to provide more accurate and targeted alarms, where the alarm generator will also take into account various context (*i.e.*, physiological, diagnostic, environmental) in order to create a context-aware clinical picture of a patient and produce high quality alarms [28], [27]. Figure 4 shows how medical devices are interconnected in an example system for Blood Glucose (BG) control [27], which monitors a patient's BG level, alerts clinicians *e.g.*, in case of hypoglycemia and/or adjusts the insulin infusion rate for the patient.

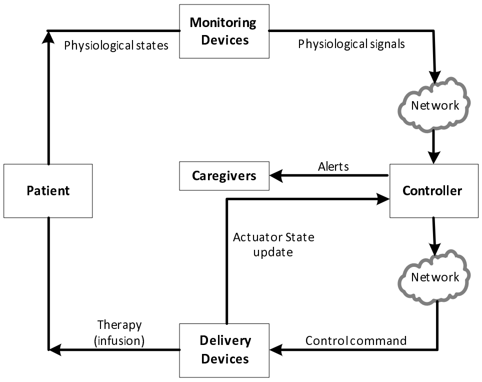

Figure 4. A Blood Glucose regulation MCPS

Patient data managed by a MCPS are considered safety-critical information that needs to be protected from failure or adversarial behavior. Apart from typical access control requirements for establishing confidentiality and integrity [27] against attacks/failure that may target the patient's privacy and/or health state, the availability of communicated data in a MCPS system is also very important, although often neglected. In the BG control system for example, if a "stop" control command fails to reach the infusion pump due to network congestion or failure the result is life-threatening.

Real-time availability. We assume that the MCPS system is run over a centrally managed trusted network and that a TC interoperability architecture (*e.g.*, the Trusted Network Connect [24]) is used. For trusted access control we use a TPM interface. This can be combined with the Kerberos Authentication Service [29] for secure multicasting. All devices are TPM-compliant and support integrity and confidentiality guarantees against active adversaries, and in particular DoS attacks. External access to the network is via a gateway that will only forward packets that are authenticated. To prevent DoS attacks all IP packets must be authenticated and linked to authorized users. Users have a limited bandwidth, and IP packets must be of a specific format/type. If necessary, Intrusion Detection/Prevention mechanisms are used.

Availability in the MCPS can be impacted by malicious actions that can lead to the network traffic being congested. In such cases the congestion level is raised and hi priority packets (*e.g.*, smart alarm data or a 'stop' command in a BG control system) are serviced first, while packets of lower priority (e.g. originated by a monitoring device) may be dropped. The above policy guarantees real-time availability under the same reliability assumptions as in Section IV.

VI. Conclusions and Future Work

An extension of the ABAC model was presented in which real-time attributes take into account the priority of access requests to support real-time availability against an active adversary. We have shown how our model can be used in CPS applications that are protected by a TC architecture. While we focused on network applications, the scope of real-time availability is broader and applies to real-time dynamic systems with strict time constraints imposed by physical processes. We leave this for future work.

Acknowledgement

This material is partially based upon work supported by the National Science Foundation Grant No. 1027217.

References

[1] I. Swapna, “Cyber Security for Smart Grid, Cryptography, and Privacy,” *International Journal of Digital Multimedia Broadcasting*, vol. 2011, 2011.

[2] Y. Mo, T. Kim, K. Brancik, D. Dickinson, H. Lee, A. Perrig, and A. Sinopoli, “Cyber-Physical Security of a Smart Grid Infrastructure,” *Proc. IEEE*, vol. 100(1), pp. 195–209, 2012.

[3] T. Fleury, H. Khurana, and V. Welch, “Towards a Taxonomy of Attacks Against Energy Control Systems,” in *Critical Infrastructure Protection II*, vol. 290, pp. 71–85, 2009.

[4] A. Cárdenas, S. Amin, Z. Lin, Y. Huang, C. Huan, and S. Sastry, “Attacks Against Process Control Systems: Risk Assessment, Detection, and Response,” in *6th ACM Symp, on Inf. Comp. and Comm. Sec.*, ACM, pp. 355–366, 2011.

[5] M. Burmester, E. Magkos, and V. Chrissikopoulos, “Modeling Security in Cyber-Physical Systems,” *Int. Jour. Crit. Infr. Protect.*, vol 5(3-4), Elsevier, pp 118–126, 2012.

[6] M. Blanke, M. Kinnaert, J. Schröder, and J. Lunze, *Diagnosis and Fault-Tolerant Control*, Springer-Verlag, 2006.

[7] E. D. Bell and J. L. La Padula, “Secure Computer System: Unified Exposition and Multics Interpretation,” Bedford, MA, 1976, http://csrc.nist.gov/publications/history/bell76.pdf

[8] K. J. Biba, “Integrity Considerations for Secure Computer Systems,” MITRE Corp., Tech. Rep., 1977.

[9] D. Latham, “Department of Defense Trusted Computer System Evaluation Criteria,” *Department of Defense*, 1986

[10] R. S. Sandhu and P. Samarati, “Access Control: Principles and Practice,” *IEEE Communications Magazine*, vol. 32(9), pp. 40–48, 1994.

[11] D. F. Ferraiolo, S. I. Sandhu, R. S.and Gavrila, D. R. Kuhn, and R. Chandramouli, “Proposed NIST Standard for Role-Based Access Control,” *ACM Transactions on Information Systems Security*, vol. 4(3), pp. 224–274, 2001.

[12] E. Yuan and J. Tong, “Attributed Based Access Control (ABAC) for Web Services,” in *Proceedings of the 2005 IEEE International Conference on Web Services (ICWS 2005)*. IEEE, 2005, pp. 561–569.

[13] R. S. Sandhu, E. J. Coyne, and C. E. Feinstein, and H. L. Youman, “Role-Based Access Control Models,” *IEEE Computer*, vol. 29(2), pp. 38–47, 1996.

[14] D. F. C. Brewer and M. J. Nash, “The Chinese Wall Security Policy,” in *IEEE Symposium on Security and Privacy*, pp. 206–214, 1989.

[15] E. Bertino, P. A. Bonatti, and E. Ferrari, “TRBAC: A Temporal Role-Based Access Control Model,” *ACM Transactions on Information and System Security (TISSEC)*, vol. 4(3), pp. 191–233, 2001.

[16] J. Joshi, E. Bertino, U. Latif, and A. Ghafoor, “A Generalized Temporal Role-Based Access Control Model,” *IEEE Transactions on Knowledge and Data Engineering*, vol. 17(1), pp. 4–23, 2005.

[17] T. Priebe, E. Fernandez, J. Mehlau, and G. Pernul, “A Pattern System for Access Control,” *In Research Directions in Data and Applications Security XVIII*, Springer US, pp. 235–249, 2004.

[18] T. Priebe, W. Dobmeier, C. Schläger, and N. Kamprath, “Supporting Attribute-Based Access Control in Authorization and Authentication Infrastructures with Ontologies,” *Journal of Software*, vol. 2(1), pp. 27–38, 2007.

[19] S. Blake, D. Black, M. Carlson, Z. Wang, and W. Weiss, “An Architecture for Differentiated Services,” *RFC 2475*, Dec. 1998.

[20] R. Shenker, S. Braden and D. Clark, “Integrated Services in the Internet Architecture: an Overview,” *RFC 1633*, 1994.

[21] R. Braden and L. Zhang, “Resource ReSerVation Protocol (RSVP)–Version 1 Message Processing Rules,” RFC 2209, 1997.

[22] International Electrotechnical Commission, “IEC/TR 61850-90-5, Edition 1.0 2012-05, Technical Report, Power Systems Management and Associated Information Exchange – Data and Communications Security,” *http://webstore.iec.ch/preview/info_iec61850-90-5%7Bed1.0%7Den.pdf*, May 2012

[23] TCG, “TPM Structures, Level 2, Version 1.2, Revision 116, Communication Networks and Systems for Power Utility Automation,” *http://www.trustedcomputinggroup.org/resources/tpm_main_specification*, March 2011.

[24] TCG, “Trusted Network Connect Architecture for Interoperability; Specification 1.3; Revision 6,” April 2008.

[25] D. Guidry, M. Burmester, X. Yuan, X. Liu, J. Jenkins, and S. Easton, “Techniques for Securing Substation Automation Systems,” in *7th Int. Workshop on Crit. Inform. Infrastr. Secur. (CRITIS 2012)*, 2012.

[26] SISCO, “Cisco and SISCO Collaborate on Open Source Synchrophasor Framework, Press Release,” Tech. Rep., 2011.

[27] I. Lee, O. Sokolsky, S. Chen, J. Hatcliff, E. Jee, B. Kim, A. L. King, M. Mullen-Fortino, S. Park, A. Roederer, and K. Venkatasubramanian, “Challenges and Research Directions in Medical Cyber-Physical Systems,” *Proc. IEEE*, vol. 100(1), pp. 75–90, 2012.

[28] M. Imhoff, S. Kuhls, U. Gather, and R. Fried, “Alarm Algorithms in Critical Care Monitoring,” *Anesthesia and Analgesia*, vol. 102(5), pp. 1525–1536, 2006.

[29] C. Neuman, T. Yu, S. Hartman, and K. Raeburn, “The Kerberos network authentication service (V5),” *RFC 4120*, 2005.