Formal Specification of Security Policies in Multi-Domain Networks

Devrim Ünal*, M. Ufuk Çağlayan**

January 17, 2009

Abstract

Security management for integrated land-mobile networks requires security policy management on multiple domains. Mobile users present a challenge for specification and verification of security policies in this environment. Formal methods ensure that a system's construction adheres to its specification. Formal methods for specification and verification of security policies ensures that in given network configuration the security policy is consistent and satisfied by the network elements. We present a framework for specification and verification of security policies based on formal methods of model checking proving. Our framework is accessible by system administration personnel and formal specifications are developed and verified in an incremental and automatic manner without the need of formal methods knowledge by the user.

Özet

Entegre yer ve hareketli ağlarda güvenlik yönetimi çoklu etki alanlarında güvenlik yönetimini gerektirir. Hareketli kullanıcılar bu ortamda güvenlik politikalarının betimleme ve doğrulamasında zorluk arzetmektedir. Formal yöntemler bir sistemin betimlemesine uygun olduğunu ispat etmekte kullanılırlar. Güvenlik politikalarının betimlenmesi ve doğrulanmasında kullanılan formal yöntemler bir ağ yapılandırmasında, güvenlik politikasının tutarlı olduğunu ve ağdaki öğeler tarafından sağlandığını ispat eder. Bu çalışmada model denetleme yöntemiyle güvenlik politikalarının betimlenmesi ve doğrulanması ele alınmaktadır. Çerçevemiz sistem yönetimi personeli tarafından erişilmekte olup formal betimlemeler sistem tarafından artımsal ve otomatik bir şekilde kullanıcının formal yöntemler bilgisi gerekmeden üretilmesini sağlamaktadır.

Anahtar Sözcükler— Model doğrulama, ağ güvenliği, güvenlik politikası

* Devrim Ünal is with the National Research Institute of Electronics and Cryptology (e-mail: devrimu@uekae.tubitak.gov.tr).

** Prof. Ufuk Caglayan is with the Bogazici University Computer Engineering Department. (e-mail: caglayan@boun.edu.tr).

This work is supported by the Turkish State Planning Organization (DPT) under the project number 2007K120610.

1 Related Work

The logic based approach of specifying information security policies is based on universal constructs and capable of supporting formal calculus methods. Use of logics also enables automated tool support for model checking and theorem proving.

The Flexible Authorization Framework (FAF) [1], [2] is a logic programming based method for definition, derivation and conflict resolution of authorization policies. Another significant study based on logic that supports explicit denials, hierarchies, policy derivation and conflict resolution is [3]. Woo and Lam [4] define a paraconsistent formal language for authorizations based on logical constructs. In [5] deontic logic is used for modeling the concepts of permission, obligation and prohibition with organizational concepts such as responsibility, delegation and time constructs. A security policy language based on the set-and-function formalism is presented in [6].

Reasoning about spatial configurations for application level security policies in ubiquitous environments is one of the issues investigated in [7]. In this study a simplified version of ambient calculus and ambient logic is used in policy rules of a security policy. Process calculus based security policy specification based on the Sπ calculus that uses Datalog for implementation is presented in [8].

Similarly, in our approach, process calculus and its modal logics are utilized. In contrast to [8], the Ambient Calculus [9] will be used with implementation in an automated theorem proving tool. In contrast to [7], network level policies rather than application level policies will be covered and location relative to domains and hosts will be modeled rather than physical location.

2 Description of Case Study

In this study we assume that two universities connected over the Internet provide common services for the students, research assistants and lecturers. These users are mobile and they can roam in the network between domains, access resources from another domain or the Internet. There are three services provided, Internet connection, Library facilities and Joint Project Access. The Internet connection is provided for students of other university by wireless access. Members of both universities may browse catalog of the Libraries, lend books and access online resources based on their access rights. The joint project requires information access by faculty members from each organisation into a shared server. The project also involves some research assistants and students enrolled in the both universities to locally log in to university networks and do some work.

In 2.1 we give the scenario for inter-domain communication involving mobility between domains. In 2.2 we present a written description of how to write an example Inter-Domain Security Policy which includes the notion of location constraints. A good reference in this subject is the recommendations of NIST [REF] on interconnection of systems and adds . The security policy in this context is called an Interconnection Security Agreement (ISA). Herein we are interested about the service access and information exchange over the interconnection. Therefore we name the interconnection scenario as "Inter-Domain" scenario.

2.1 Inter-Domain Scenario with Mobility

The University A and University B are connected via the Internet (Public Network). The interconnection discussed herein allows users of both universities to exchange information and use specified information technology resources of one another from their respective campuses.

In each university, lecturers, graduate students, undergraduate students and research assistants are known with a unique identity. We assume a common authentication infrastructure which allows inter-domain authentication of the users via a common protocol such as RADIUS or DIAMETER. Another alternative is that the two universities share a common operating system environment where each domain trusts the other domain. The users can be connected to the network of either university or the Internet and they authenticate to their home domain (which is the main network they are registered).

Once users are authenticated they are free to roam between domains. Users may access information and services in either university from either campus or the Internet based on a security policy. The security policy restricts the usage of certain services and access to information based on the permissions of the users.

2.2 Inter-Domain Security Policy

The information technology facilities that will be used from other campuses are, the library, and technology development centers residing at both campuses. The libraries allow lending books and periodical to lecturers and students of both institutions. The libraries have an on-line information system for the borrowers to search, reserve publications and get information on their borrowing status, as well as access

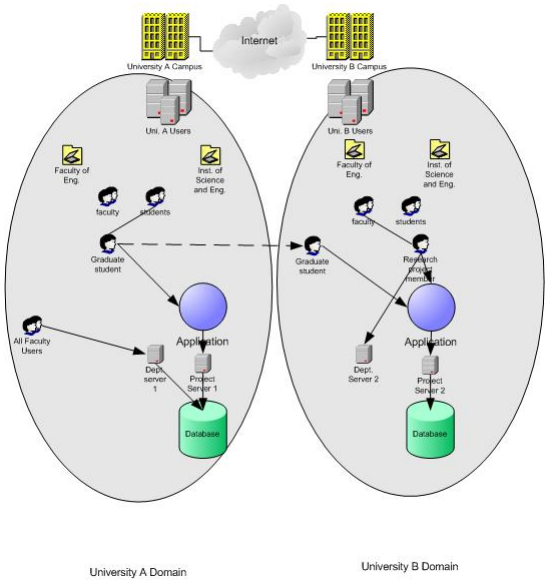

Figure 1: University Inter-Domain Scenario with Mobility

on-line publications subscribed by that library. The technology development center hosts joint projects whereby students and researchers from either institution participate as project members. The project members have a need to access and share information both locally and remotely from both campuses. The members of each university will also be allowed to access the Internet through wireless LAN access available in both universities.

Information relating to the joint research projects between two universities, and the information within the online library system will made available by the interconnection of the two systems. The joint research project information consists of a database, a web application that controls access to the database, project files, report files and plan files. The Library information consists of periodicals and online publications at other organisations that are accessible by on-line subscription, catalogue information about hard-copy publications that reside in the library and Lending / borrowing information for hard copy publications.

For joint research project information, bi-directional information exchange, local access by project members from both universities and remote access to other campuses will be provided, For the online library system, local access and remote access over the Internet will be provided. The user community involved in the scenario is, Lecturers, Research Assistants, Graduate Students and Undergraduate students.

The access rights are then determined for each role for every service. The type of access will be based on the rights of the roles. The last information that is included are the location constraints. This is a novel characteristic of our security policy model that location constraints can be specified. For the use of a service, the activation of an activity or the use of a resource, the security administrator can specify location constraints. The set of locations is predetermined for each case study. For our case study, we allow the following locations: Internet, Local, Remote and Other Domain.

3 A Model for Multiple Domain Mobile Networks

3.1 Network Model

The network model defines a multi-domain mobile network consists of four sets: administrative domains, hosts, users and objects.

1. Administrative domains (D): An administrative domain defines a certain mobile network. An administrative domain is also a container for other network elements, i.e. it defines a set of subjects, objects and security policy for a domain.

2. Hosts (H): Hosts are computing terminals and a container for objects. Every user needs to be logged onto a host in a domain to be able to access objects in the system.

3. Users (U): The users may be member of a single domain as home domain. They may also roam from one domain to another. We assume that the user is known with the same identity throughout all the domains.

4. Objects (O): The objects are resources in a domain. Objects model communication ports, databases, files, and messages. Objects are contained in hosts, and may enter or leave hosts and domains. Messages may traverse the network.

We can represent the mobile network model elements graphically as in Fig. 2. This model is based on the Ambient Calculus with addition of security policy. The circles in Fig. 2 represent basic system elements, domains, hosts, users and objects. The label of a domain represents security policy. We assume that hosts, users and objects do not define their own security policy; they are governed by the security policy of domains. In this example system model, Host H1 is a server, H2 is a client logged onto this domain and H3 is a portable host not logged onto any domain. We assume that all the elements reside in a system element called the World.

3.2 Mobility Model

In 3 a graphical representation for a multi-domain mobile network architecture is given. In this model we are interested in formal specification of the mobility of hosts, users and objects as ambient calculus processes. In our mobility model, hosts, users and objects have the following mobility capabilities:

Hosts: Moving into a domain represents connecting a host to a domain. Moving out represents disconnecting.

Users: If users move into a host, this represents logging into a host. If a user moves out of a host, the user is logged out. In this case the user is removed from active users set of the domain.

Objects: Every object must reside in a host when not on the move. An object that moves from one host to another is called a message and movement of messages represents communication.

4 Security Issues In Multi-Domain Mobile Networks

In this section we discuss what kind of security breaches arise if security policies for collaboration between domains (or inter-domain security policies) are not checked and enforced globally by the system. We assume that some kind of written inter-domain security policy exists. The examples presented will show how this policy is breached because of mobility.

In the first example, Domain B is a child-domain of Domain A. This may be the case where Domain A has a higher classification level than Domain B. Domain A users are also Domain B users and may roam between domains, but may be logged onto one domain according to the classification level of the objects that they work on. The written inter-domain policy of Domain A and Domain B states that Domain A files are not allowed to be read from Domain B. Domain A users are allowed to read Domain A and Domain B files. All Domain A users may read and write the object O2 (if logged onto Domain A).

Let's assume that a user U1 of Domain A is logged onto Domain B. U1 moves (with U1's associated host H1) into Domain A. U1 may then read O2 since the domain policy allows active Domain A users to conduct this action. This read may be followed with a copy to U1's own host since this host is under control

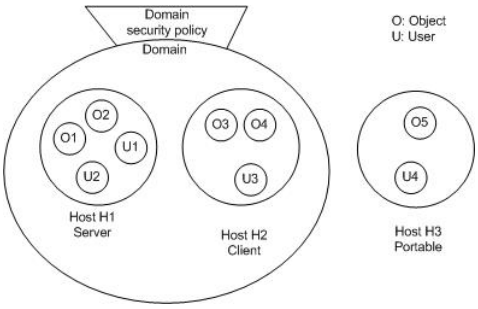

Figure 2: A mobile network with single domain, different hosts, users and objects

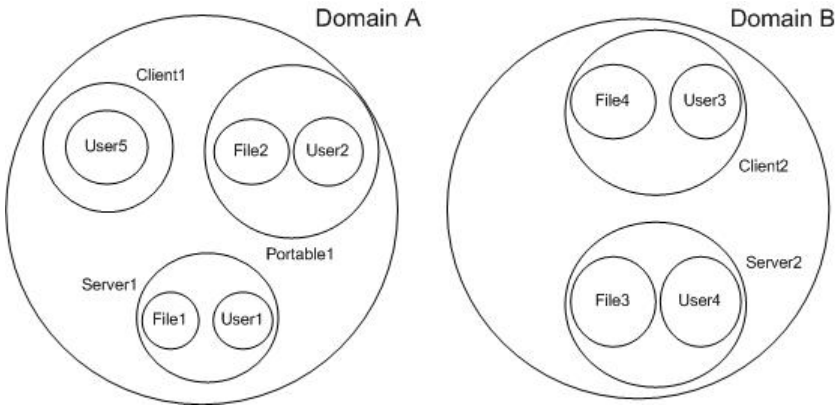

Figure 3: Graphical representation of a multi-domain network.

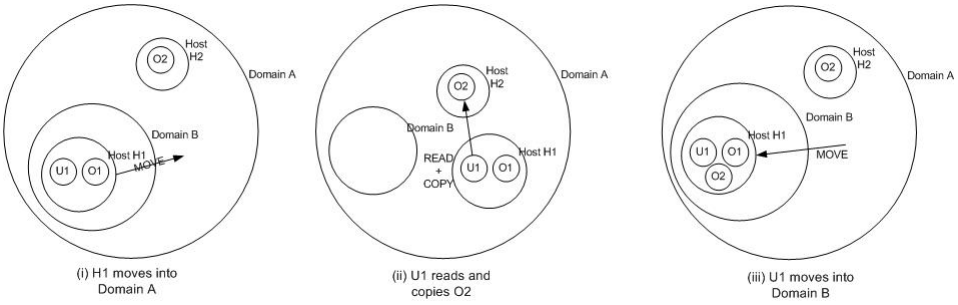

Figure 4: Example of a security breach

of the user. Then U1 moves back into Domain B. U1 can export O2 into Domain B, which is a breach of total system policy. It should be noted that U1 has performed actions allowed by the domain policies. The main reason for such breaches is that the system does not track and enforce the inter-domain policy.

Another possible way for the same kind of breach to happen is that, an active user of Domain A (Let's say U2) can read O2, move into DA, copy O2 to another file O3= O2. Then U1 can read O3 (or equivalently O2) which is another violation of the overall policy.

Another scenario is as follows. We assume two domains, Domain A (DA) and Domain B (DB). The domain policies for both domains allow all domain users to read all objects in the domain. The inter-domain policy does not allow access from DA to DB, but allows communication from DB to DA. The intention is that DB users may access DA (and information can flow from DA to DB), but not vice versa. This inter-domain policy may be enforced by the outer firewalls of DA and DB, which realize the domain security mechanisms Assume the following system configuration (here we use a textual representation):

Active Hosts in DomainA = AHDA{H1, H2, H5}

Objects in and Users logged onto H1 = OUH1 = {O1, U1}, OUH2 = {O2,U2} OUH5 = {U5}. U5 is administrator for DA.

For domain B, AHDB= {H3, H4}. OUH3 = {O3, U3}, OUH4={O4,U4}

Action sequences:

1. U3 reads O3, U3 sends O3 to U1. Information flow constraints of the overall policy has been breached in this instance.

2. U3 reads O3 and O4, U3 moves to DA. U2 can read O3 and O4 since U3 is now an active user of the DA domain and U2 can read objects in DA. This is another breach of the intended information flow.

3. Executable file breach: Assume that users can only execute certain files in D2. The following sequence of actions lead to a breach of this rule: U3 moves to D1 together with H3, U3 reads copies executable O1 to its host H3. U3 moves to D2 and executes O1.

4. Administrative breach: DB user H3 moves to DA. A user in DA with Domain Administrator rights for DA (U5) can read O3. If H3 had not moved, U5 would not be allowed to read O3. U5 as an administrator for DA, gains the capability to administer DB hosts that log on to DA because of roaming.

Figure 5 shows a scenario involving multiple domains. Domain A and Domain B have an interconnection. Domain B previously had an interconnection with Domain C. Assume that according to this interconnection H3 is accessible and readable by Domain C. Also assume that H1 in Domain A is accessible by Domain B hosts. There is no interconnection or agreement between A and C for exchange of information, therefore A objects can not be read by C subjects. The following sequence of actions lead to an unintended information flow: U2 can read O1, copy or write O1 as O3 in Host H3. This information can be read by U4 of Domain C.

These kind of breaches originate from the following general weakness in contemporary systems:

Inter-domain actions (such as communication, access) are generally checked by using firewalls which are at the border of a domain. These mechanisms can check inter-domain actions when they happen from outside the domain. Once a user of another domain B enters into domain A a firewall is no longer effective.

The host and intra-domain security mechanisms are usually set-up in a way that all objects and subjects within a domain are known and trusted (employees of an organization, students of a university etc. which have liabilities against their organization). However when inter-domain roaming is allowed, users outside of an organization's responsibility may conduct actions inside. Those users (and/or their hosts) are subject (or configured) to adhere to their own organization's security policy. This policy may not match the visited institution's policy. If there is no enforcement of inter-domain security policy within a domain, the overall policy enforcement may be endangered.

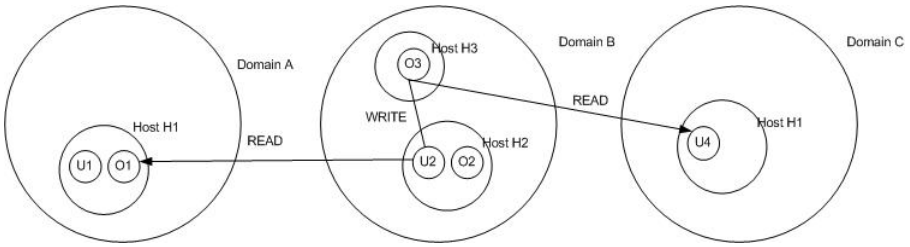

Figure 5: Multiple domain scenario

The movement of hosts may conceal movement of objects. This movement could breach information flow security policies. A method of tracing such movement is by keeping traces of events, but such traces may be hard and time-consuming to analyze once a long time passes in the system. A more optimal approach that is taken here is to keep the current state of the system, that shows the location and movement of objects.

Administrative weakness is inherent in modern operating systems such as Windows and Unix. The user with the root or domain administrator credentials can virtually access any resource within a network or domain. While this may be acceptable for a static world where all users and objects within a network or domain stays within those network or domain, it may cause some serious security problems when inter-domain collaboration and movement is in place. A user's host may carry information not normally open to access by other organizations, but by moving and logging onto a domain, the user's host becomes administrable by the foreign domain administrator. Objects on a roaming user's host become accessible to foreign administrators.

Multiple interconnections: When an organization (A) interconnects with another, this is mostly seen as a bi-directional information flow. One of the organizations (B) could have an interconnection with another (C); associated with permissions based on the requirements at the time of this interconnection. So-called 'back-end interconnection' from the point of view of A, the permissions arising from B-C interconnection may breach A's security policy.

5 Formal Specification

5.1 Formal Specification of Security Policy

The access control model is specified using Predicate Calculus and First Order Set Theory. The system model is based on the RBAC model. We use the hierarchical RBAC model that supports role hierachies. To the Hierarchical RBAC model we add Domains, Hosts and Object Types. We remove the concept of sessions and assume that sessions are dynamically changeable with changes in the system model.

Constants:

d, n, m, o, s, t: Number of domains, hosts, users, roles, user groups and objects, respectively.

Sets:

$D = \{D_a, D_b, D_c..., D_z\}$: Set of domains

$H = \{H_1, H_2, ...H_n\}$:Set of hosts

$U = \{U_1, U_2, ...U_m\}$: Set of users

$R = \{R_1, R_2, ...R_o\}$: Set of roles

$O = \{O_1, O_2, ..., O_t\}$:Set of objects

$OT = \{OT_1, OT_2, ...OT_v\}$: Set of object types. For this study we take $OT = \{Application, File, Network, Database\}$

Relations:

$HOD : H \times D$: Relation mapping hosts to domains. $HOD(H_i, D_a)$ denotes that H_i is enrolled to Domain D_a.

$UOD : U \times D$: Relation between users and domains. $UOD(U_j, D_a)$ denotes that U_j is enrolled to Domain D_a.

$OOT : O \rightarrow OT$:Function that specifies the type of an object. $OOT(O_k)$gives the type of object O_k.

$UA : U \times R$: Relation for assignment of users to roles.

The authorization matrices are completely user definable. They are associated with an XML schema and the input and checking of the matrices is established with a GUI. Here we give the mathematical description of the matrices and set of values involved in defining these matrices. The model we use for services, activities and access types enable the inclusion of Service Oriented Architectures and Web based services in our framework. Use of Locations enables the specification of mobility constraints. Therefore the framework becomes suitable for specification and verification of mobile, service based computing infrastructures.

Sets:

$SERV$: User definable set that defines the services available within an inter-domain communication scenario. The following example defines a set of services available within universities for Internet access, use of Library Resources and Joint Project access for collaboration in research projects. $SERV = \{Internet, Library, JointProject\}$

ACT: User definable set which represents the activities that take place for successful operation of Services.

$ACTSER : ACT \times SERV$: For each Service there is a set of activities involved. Therefore ACTS is a relation that defines the mapping of activities to services.

A : Set of actions conductable by subjects on objects. We take A to be fixed in this study:

A={Connect, Login, Logout, Manage, Execute, Read, Write, Delete, Create}

L : Set of locations that accesses are allowed from. In this study we define certain locations. $L = \{Local, LocalWireless, OtherDomain, Internet\}$

Service Access Matrix : $SAM : UG \times SERV$, $SAM[UG_i, SERV_j] = True \mid False$. If SAM[i,j] = True then User Group[i] is allowed to use Service [j].

Activity Matrix: $ACM : UG \times R \times ACT$. $ACM[UG_i, R_j, ACT] = True \mid False$. If ACM [i,j,k] = True then User Group [i] is allowed to conduct Activity [k] only if assuming Role [j] i.e. UA(User Group[i],Role[j]) is defined. For each service there is an Activity matrix that includes activities in $ACTS$ for that service.

Access Type Matrix: $ATM : UG \times R \times A$ defines the matrix that specifies allowed actions to be conducted by specific User Groups assuming certain Roles. For every Service there is one Access Type Matrix.

Location - Activity Matrix: $LAM : UG \times R \times ACT \times LOCATION$ defines the Location constraints for a certain Activity to be conducted. $LAM[UG_i, R_j, ACTk, L_l] = True \mid False$. If LAM [i,j,k,l] = True then the ACT_k can be conducted by user group UG_i assuming role $R_j$from location L_l.

Sets:

AS: $AS = U \cup UG \cup R$. The set of Authorization Subjects. Authorization Subjects are active entities that may conduct an Action on an Authorization Object.

AO: $AO = O \cup OT \cup H \cup D$. The set of Authorization Objects. The authorization object is the entity upon which an action is conducted.

Signs: $S = \{+, -\}$Represents permission or denial.

Signed Actions: $S \times A$: Represents permission or denial of an action (+,read) denotes that read action is permitted.

Predicates

1. EnrolledDomainHost (*host*, *domain*): True if *host* is a registered member of the Domain *domain*.
2. EnrolledDomainUser (*user*, *domain*)
3. ActiveDomainUser (*user*, *domain*)
4. RoleAllowed (*user*, *role*)
5. ActionAllowed (*as*, *action*): True if Authorization Subject *as* is allowed to execute action *action*.

Conditions: First-order sentences built on the Predicates defined above.

Spatial Formula: Ambient Logic formula that includes names of domains, authorization subjects and authorization objects.

5.2 Formal Specification of Mobility

The formal model for mobility is a finite fragment of the ambient calculus with public names as used in [X] and shown in Table X. The open capability is only used for messages. The reason for this restriction is that "opening" other system elements such as hosts would violate the integrity of the model.

In the formal specification, domains, hosts, users and objects are modeled as Ambients. The actions are modeled as Ambient Calculus capabilities. A process specification shows a trace of a process in a certain mobile network scenario. Each scenario may be modelled as a set of process specifications. These specifications will then be checked against a security policy for compliance.

Below some examples of object, host and user mobility specification of mobility as ambient calculus processes are listed. Some known notation conventions are utilized: for example n[] means n[0]. The symbol → represents the reduction relation and →* represents a series of reductions.

1. Object Mobility

   File 1 is copied to Portable 1: World[DomainA[Server1[folder [out folder. out Server1. in Portable1. in folder. File1[] | File1 [[]] | Portable1[folder[]]]] →* World[DomainA[Server1[folder [File1[]]] | Portable1[folder[File1[]]]]]

   File 2 is deleted from Portable1. (moved into the Trash can in Portable1): World[DomainA[Portable1[User2[] | folder [out folder. in Trash. File2[]]|Trash[]] →* World [DomainA [Portable1[User2[] | folder[] | Trash[File2]]]]

   A message is sent from User 1 to User 3: World [DomainA [Server1 [User1 [message[M | out User1. out Server1. out DomainA. in DomainB, in Client2. in User3.0]]]] | DomainB [Client2 [User3 [open message.(m).0]]] →* World [DomainA[Server1[User1[]]] | DomainB[Client2[User3[M]]]]

2. Host mobility

   Portable 1 disconnects from Domain A and connects to Domain B: World [DomainA[Portable1[out DomainA. in DomainB .0]] | DomainB[]] → World [DomainA[] | Portable1[in DomainB. 0] | DomainB[] → World [DomainA[] | DomainB[Portable1[]]]

3. User mobility

   User 1 logs into Server 1: User1[in Server1.0]|Server1 [File1[]] → Server1 [User1[] | File1 []]

   User 1 logs off from Server 1: Server1 [User1[out Server1.0] | File1 []] → User1[]|Server1 [File1[]]

Conclusion

We have presented a method for specification and verification of complex security policies for multi-domain mobile networks. Our approach stems from the formalisms of ambient calculus and logic based authorization frameworks. The contributions of this study

are: (i) flexible policy specification for process calculus models, (ii) a formal inter-domain security policy model, (iii) mobility and location based security policy specification.

References

1. Jajodia, S., Samarati, P., Subrahmanian, V. S., A Logical Language for Expressing Authorizations, Proceedings of the 1997 IEEE Symposium on Security and Privacy, IEEE (1997) 31-43

2. Jajodia, S.,"Flexible Support for Multiple Access Control Policies", ACM Trans. Database Systems, Vol. 26, No: 2, (2001) 214-260.

3. Bertino, E., Ferrari, E., Buccafurri, F., and Rullo, P, A Logical Framework for Reasoning on Data Access Control Policies. In Proceedings of the 1999 IEEE Computer Security Foundations Workshop. CSFW. IEEE Computer Society, Washington, DC, 175 (1999).

4. Woo T. Y. C. and Lam S. S., Authorizations in distributed systems: A new approach. Journal of Computer Security, 2 (1993) 107–136.

5. Cuppens, F., Saurel,C., Specifying a Security Policy: A Case Study, 9th IEEE Computer Security Foundations Workshop, Kenmare, Ireland, IEEE Computer Society Press, (1996) 123-134.

6. Ryutov, T., Neuman, C., Representation and Evaluation of Security Policies for Distributed System Services, Proc. DARPA Information Survivability Conference, DARPA (2000)

7. Scott D.J., Abstracting application-level security policy for ubiquitous computing. UCAM-CL-TR-613, Cambridge University (2005)

8. Fournet, C., Gordon,A.D., Maffeis,S., A Type Discipline for Authorization Policies, Lecture Notes in Computer Science, Volume 3444. Springer-Verlag,(2005) Pages 141 – 156

9. Cardelli, L., Gordon, A.D., Mobile Ambients, Theoretical Computer Science 240 (2000) 177-213