Evaluating Security Policies in Pervasive Mobile Environments Using Context Information

Carlos Sánchez[1], Le Gruenwald[1], and Mauricio Sánchez[2]

[1] The University of Oklahoma, School of Computer Science, Norman OK USA
{maletas,ggruenwald}@ou.edu
[2] IMTEC Corporation a 3M Company, Ardmore OK USA
mauricio.sanchez @imtec.com

Abstract. Due to both the number of entities and the nature of the interactions and collaborations amongst them, conventional security models are inadequate for regulating access to data and services in a pervasive mobile computing environment. Since many of these interactions occur between entities that have not interacted with each other previously, new security paradigms rely on context information in order to arrive at a security and decision. However, these new systems fail to take into account the variability, correlation and uncertainty of the context variables composing a security policy when making a security decision. In this paper, we propose a Monte Carlo based framework to evaluate security policies that are based on the changes in multiple context variables. In this framework, context variables are modeled and risk in security decisions is measured.

Keywords: Security, integrity and protection, risk, context data, pervasive mobile environments.

1 Introduction

With ever increasing number of information sources, a modern collaboration system needs to not only rapidly assemble a set of disparate information systems into a coherently interoperating whole, but also make sure that the interactions amongst different entities participating in the system are secure. That is, strong security measures must be available to enforce data integrity and protect sensitive information. In addition, collaboration systems must take into account the environment in which not only no or few fixed infrastructure nodes exist, but also access to information depends on some context information (i.e. location) by the requesting entity. Because of the lack of fixed infrastructures and the variability of context data, it is unreasonable to expect that every entity in the collaboration system stores not only a reference (identity) to but also relevant context data relating to any other entity it may interact with as a way to enforce access control. It is for these reasons that traditional security implementations, such as access control lists (ACL) [28] and role base access controls (RBAC) [22], [23], are ill equipped to handle security operations since not only they expect the identity of the accessing entities to be known, but also they were not originally designed to support context based access. Moreover systems (i.e. [32], [36])

E. Bertino and J.B.D. Joshi (Eds.): CollaborateCom 2008, LNICST 10, pp. 145–161, 2009.

that have been expanded with context variable predicates (i.e. location, power resource, and connection type) to define security policies fail to take into account the fast changing nature of context information. This variability introduces uncertainty, which in turn introduces risk when a security decision is made. In addition, once the security decision is made, current security models do not revaluate the security decision after a specific time frame even though the context data is likely to change (i.e. the security decision would stand firm regardless of future changes in the context data) [26]. Therefore [secure] collaboration in dynamic communities that use context data to enforce access control should develop risk models to help an entity decide when access to sensitive data should be authorized.

To illustrate a secure collaboration, let us expand the scope of the interaction between a soldier and his commander in the example given in [26]. In this example, a soldier that moves about in the battlefield and dynamically retrieves enemy data from his nearby commander. In order to receive intelligence data, a soldier must first identify himself with his company commander and his platoon leader (data is restricted according to identity). The intelligence data is delivered provided that the soldier and his commanders (company and platoon leader) are within a 25-kilometer radius of a passing ScanEagle [2] UAV. Moreover, in order to avoid leaks, the platoon leader has to be within 150 meters of the company commander. To make sure that all the data is delivered, the soldier's mobile unit must have a minimum of 60% power availability. Finally, in order to receive any data, the connection rate amongst the soldiers needs to be above a specific threshold to guarantee that the communications are not being jammed (again data transfer is limited to a specific connection mode).

As the example points out, the security implementation that allows the soldier to receive the intelligence data must take into account the variability and correlation of the commanders and the UAV location information, the communication connection rate and power of his mobile unit. Thus, in order to make a reliable security decision, the security system should try to predict how much the different variables (i.e. commanders and UAV location, etc) would change by the time the enemy data is delivered to and used by the soldier, since by the time the data is delivered, any of the entities (solider, commanders or UAV) may have moved to a new place making the security decision invalid. Moreover, the security system should be able to detect, tolerate and ignore invalid context assertions made by compromised or hostile entities. Furthermore, the security implementation should be able to use a certain amount of data history for each of context variables in order to not only predict future values but also to measure the degree of change (volatility) of the context variables in order to arrive at a more accurate security decision [26]. In addition, the security system needs to determine the risk involved when making a security decision because of the changeability and correlations of the context variables. Finally, due to the dynamic nature of a context variable, each security decision should hold only for a certain degree of time.

In this paper, we extend our previous work ([26]) to model the behavior of context variables in a security policy using a random walk framework that takes into correlations amongst them. Finally, the model in [26] is further expanded to include a risk measure for the overall change in a policy according to the changes of its constituting context variables.

The remainder of this paper is organized as follows. Section 2 details a random walk framework to model context variables. Section 3 describes the Monte Carlo method used in our framework. Section 4 describes the measure of risk when evaluating a security policy. Section 5 describes a context security policy. Section 6 reviews related work in this area. Finally Section 7 concludes the paper and presents future research.

2 Modeling Context Variables

In a pervasive mobile environment, context variables of a user may change in a non-predetermined way. That is, context variables follow a stochastic process since their values may change over time in an uncertain way. Using a stochastic process model gives us a risk-measuring tool to characterize the future change in a process' value.

One stochastic process that can be employed as a risk-measuring tool is the random walk. In the random walk, forecasts for each of the context variables' future value changes - using only its past variations – can be constructed. However, before defining the random walk model we need to introduce the concepts of a context variable time series and variable return (In order to have an unified model, in this work we assume that all context variables can be modeled using a random walk even though, many context variables can display deterministic models).

Context Variable Time Series. A context variable time series is a collection of observations indexed by the time of each observation. An entity collects the context variable data beginning at a particular time (e.g. t=1) and ending at another (e.g. t=N). Formally, a time series for context variable a context variable V is represented in a vector form as follows [11], [13]:

$$\{V_t\}_{t=1}^{N} = (V_1, V_2, \ldots, V_i, \ldots V_N)$$

where Vi is the observation made at time I and VN is the observation made at time N.

One-Time (Single Period) Value Return Horizon. The change in the value of a variable can be expressed in a variety of forms, such as, absolute value change, relative value change, and log value change. When a value change is defined relative to some initial value, it is known as a return [13]. That is, the changes over time of a variable's value can be measured and modeled in terms of continuously compounded returns (log value changes). Table 1 shows the definition of Absolute, Relative and Log value change for a context variable V between time t and t-1 (denoted as V_t and V_{t-1}).

Table 1. Definitions of absolute, relative and log changes of a variable

Absolute value change	Relative value change	Log value change (return)
$D_t = V_t - V_{t-1}$	$R_t = \dfrac{V_t - V_{t-1}}{V_{t-1}}$	$r_t = \ln(1+R_t) = \ln\left(\dfrac{V_t}{V_{t-1}}\right) = (v_t - v_{t-1})$ where vt = ln(Vt)

Random Walk Model for One Context Variable. A random walk is a formalization of the idea of taking successive steps, each in a random direction. It may be thought of as a model for an individual walking on a straight line who, at each point of time, takes one step either to the right or to the left with different probabilities [34]. Random walk models have been applied in several fields. For instance, in economics, a random walk is used to model shares prices and other factors and in wireless networking, they are used to model node movement. Formally, a single value random walk model for a context variable can be stated as follows.

$$V_t = \mu + V_{t-1} + \sigma_t \varepsilon_t , \varepsilon_t \sim IIDN(0,1) \tag{2.1}$$

or

$$V_t - V_{t-1} = \mu + \sigma_t \varepsilon_t , \varepsilon_t \sim IIDN(0,1) \tag{2.2}$$

where IID stands for “identically and independently distributed”, and N(0,1) stands for the normal distribution with mean 0 and variance 1. That is, at any point in time the current value Vt depends on one fixed parameter μ (mean), one time based parameter σt (standard deviation), the last period’s value Vt-1, and a normally distributed random variable εt. The assumption that context values are normally distributed is helpful because 1) we only need the mean and variance to describe the distribution, 2) the sum of multiple normal context variables is also normally distributed, and 3) normality is the central assumption of the mathematical theory of errors [33].

Since returns not only have more attractive statistical properties than values, but also are often preferred to absolute value changes because the latter do not measure changes in terms of the given values [13], it is better to model the log value Vt (Table 1) as a random walk with normally distributed changes, that is:

$$v_t = v_{t-1} + \mu + \sigma_t \varepsilon_t , \varepsilon_t \sim IIDN(0,1) \tag{2.3}$$

Therefore, since we are modeling log changes, the expression for values is simply obtained by taking the inverse of the logarithm (ex), that is

$$V_t = V_{t-1} e^{\mu + \sigma_t \varepsilon_t} , \varepsilon_t \sim IIDN(0,1) \tag{2.4}$$

In [13] three core assumptions are made. First of all, returns in different periods are not auto correlated. Secondly, the variance of the returns (volatility) scales with time (it remains constant for different time horizons). Finally, the random walk model is relaxed by assuming that log values have a mean μ set to zero (in [15], it is shown that for short horizon periods, the volatility is much larger than the expected return; thus, the forecast of the future return distribution is dominated by the volatility estimate. In other words, when dealing with short horizons, using a zero expected return assumption is as good as any other mean estimate). Therefore, the model can be represented as:

$$v_t = v_{t-1} + \sigma_t \varepsilon_t , \varepsilon_t \sim IIDN(0,1) \tag{2.5}$$

in terms of returns

$$r_t = \sigma_t \varepsilon_t, \ \varepsilon_t \sim IIDN(0,1) \tag{2.6}$$

The standard deviation (volatility) at time t - σ_t - can be computed using an Exponential Moving Average (EMA) [6] of past observations, that is, for a given set of K (where K=t-1) returns, the variance can be defined as follows:

$$\sigma_t^2 = \frac{\sum_{i=1}^{K} \alpha^{i-1} (r_{t-i})^2}{\sum_{i=1}^{K} \alpha^{i-1}} \tag{2.7}$$

where the parameter α (0 < α <=1) (***smoothing factor***) determines the relative weights applied to the observations (returns), thus, allowing the latest observations to carry the highest weight, while still not discarding older observations entirely in the standard deviation estimate. In addition, when K→∞, and using the convergence property of the geometrics series:

$$\sum_{i=1}^{T} \alpha^{i-1} \cong \frac{1}{1-\alpha} \quad 0 < \alpha < 1$$

equation (2.7) can be rewritten as follows

$$\sigma_t^2 = (1-\alpha) \sum_{i=1}^{\infty} \alpha^{i-1} r_{t-i}^2 \tag{2.8}$$

In a recursive manner, equation (2.8) can be rewritten as:

$$\sigma_t^2 = \alpha \sigma_{t-1}^2 + (1-\alpha) r_t^2 \tag{2.9}$$

Equation (2.9) represents one time-unit (e.g. 1 minute) calculation of the variance defined over the period t-1 through t, where each t represents one time-unit. Therefore, in order to make forecasts for horizons greater than one-time unit, and taking the assumption stated previously, the variance estimate of a context variable data return for H time units is stated as follows:

$$\sigma_H^2 = H \sigma_t^2 \tag{2.10}$$

The equation above gives a simple way to calculate the volatility of H time units (e.g. hour) from the 1 time unit (e.g. minute) volatility.

Random Walk Model for Multiple Context Variables. In a context-based security policy with M context variables, the behavior of the returns of each of the context variables can be described as:

$$r_{1,t} = \sigma_{1,t} \varepsilon_{1,t}, \ r_{2,t} = \sigma_{2,t} \varepsilon_{2,t} \ \ .... \ r_{M,t} = \sigma_{M,t} \varepsilon_{M,t}$$

Since the context variables may be related to one another, we have to account for their movements relative to one another (that is, the context variables may be statistically dependent). Thus, the linear association between each pair of returns must be quantified. These movements are captured by pair-wise correlations. Therefore, the εt's should come from a multivariate normal (MVN) distribution [27] then:

$$\begin{bmatrix} \varepsilon_{1,t} \\ \varepsilon_{2,t} \\ \cdots \\ \varepsilon_{N,t} \end{bmatrix} \sim MVN\left(\begin{bmatrix} \mu_1 \\ \mu_2 \\ \cdots \\ \mu_M \end{bmatrix}, \begin{bmatrix} 1\rho_{12,t}\cdots\rho_{1M,t} \\ \rho_{21,t}1\cdots\rho_{2M,t} \\ \cdots \\ \rho_{M1,t}\rho_{M2,t}\cdots 1 \end{bmatrix} \right) \quad or\ \varepsilon \sim MVN(\mu_{G,t}, R_t) \tag{2.11}$$

where R$_t$ represents the correlation matrix of (ε$_1$, ε$_2$,… ε$_N$) and the mean and variance are represented by [13]:

$$\mu_{G,t} = \sum_{i=1}^{M} w_i \mu_i \ and\ \sigma_{G,t}^2 = \sum_{i=1}^{M} w_i^2 \sigma_i^2 + 2\sum_{i<j}\sum w_i w_j \sigma_{ij,t}^2$$

Moreover, the term σ_{ij}^2 represents the covariance between returns for the context variables i and j. We must remember that the covariance of two random variables X and Y is defined as:]:

$$\sigma_{XY}^2 = E[(X-\mu_X)(Y-\mu_Y)] = E(XY) - E(X)E(Y)$$

Since E(X)=μx and E(Y)=μy (E is the mathematical expectation), both of which are equal to zero according to our model, then the covariance is simply defined as

$$\sigma_{XY}^2 = E(XY)$$

Finally, let's recall that the correlation coefficient of two random variables X and Y can be calculated as follows [19]:

$$\rho_{XY} = \frac{\sigma_{XY}^2}{\sigma_X \sigma_Y} \backslash$$

where σ_X and σ_Y are the standard deviations of X and Y, respectively.

Using the Exponential Moving Average (EMA) [6], an expression to estimate the covariance and correlation of context variable log changes (returns) can be constructed. As such the covariance formula is defined as [19]:

$$\sigma_{XY|t}^2 = \frac{\sum_{i=1}^{T} \alpha^{i-1} \left(r_{X,t-i} - \bar{r}_1 \right)\left(r_{Y,t-i} - \bar{r}_2 \right)}{\sum_{i=1}^{T} \alpha^{i-1}} \tag{2.12}$$

3 The Monte Carlo Method

Given a set of M context variables, we use the Monte Carlo method as a computational algorithm to repeatedly construct scenarios to produce future values (equations 2.4 and 2.11) for each of different context variables. We will first introduce the Monte Carlo algorithm when M=1 to later define the general case.

4 The Monte Carlo Method for One Context Variable

Given a time series $\{V_t\}_{t=1}^{N}$ of N observations for a context variable V, the procedure to produce scenarios is to generate standard normal variates and use equation 2.4 to produce future values. The algorithm to simulate future values for one context variable is described below [26].

One stochastic process that can be employed as a risk-measuring tool is the random walk. In the random walk, forecasts for each of the context variables' future value changes - using only its past variations – can be constructed. However, before defining the random walk model we need to introduce the concepts of a context variable time

Table 2. Scenario Generation Algorithm for One Context Variable

```
Input: $\{V_t\}_{t=1}^{N}$
1. Choose the number of scenario trials T and smoothing factor $\alpha$
2. Compute the N-1 log value changes (i.e. returns) from $\{V_t\}_{t=1}^{N}$. The result of this calculation is a time series of returns of the form $\{r_t\}_{t=2}^{N}$
3. Using $\{r_t\}_{t=2}^{N}$ compute the variance series $\{\sigma_t^2\}_{t=2}^{N}$ using equation (2.7). Compute the volatility (standard deviation) at time t=N, that is $\sigma_N = \sqrt{\sigma_N^2}$
4. In order to simulate values for the next H time units (evaluation horizon), use equation (2.10) to compute the horizon volatility $\sigma_H = \sigma_N \sqrt{H}$
5. Define a time series $\{S_t\}_{t=1}^{T}$ to store the context variable simulated values.
6. For each trial k $(1 \le k \le T)$
    6.1. Compute the simulated log return R as follows:
        $R = \sigma_N \cdot \sqrt{H} \cdot Z$, where $\sigma_N \sqrt{H}$ represents the horizon volatility (used to simulate values for the next H time units (evaluation horizon) ), and Z is a generated IID normal value Z, that is, Z = ~N(0,1)
    6.2. Compute the $k^{th}$ simulated value $S_k$ as follows:
        $S_k = V_N \, exp(R)$ where $exp(x) = e^x$ (equation 2.4) and $V_N$ is the value of the context variable at time N
Output: $\{S_t\}_{t=1}^{T}$
```

series and variable return (In order to have an unified model, in this work we assume that all context variables can be modeled using a random walk even though, many context variables can display different deterministic models).

5 The Monte Carlo Method for Multiple Context Dependent Variables

In the previous section, an algorithm was derived to apply the Monte Carlo method to one context variable. For the case of multiple context variables, we need to take into account the correlation amongst the context variables when generating future values. The table below describes the modified algorithm [26].

Table 3. Scenario Generation Algorithm for Multiple Context Dependent Variables

```
Input: $\left(\{V_1\}_{t=1}^{N}, \{V_2\}_{t=1}^{N} \cdots \{V_M\}_{t=1}^{N}\right)$
1. Choose the number of scenario trials T and smoothing factor $\alpha$
2. For each context variable  time series $i$ $(1 \leq i \leq M)$
    2.1 Compute the $N$-$1$ returns from $\{V_{i,t}\}_{t=1}^{N}$. The result of this calculation is a
        time series of returns of the form $\{r_{i,t}\}_{t=2}^{N}$
    2.2 Using $\{r_{i,t}\}_{t=2}^{N}$ compute the variance series $\{\sigma^2_{i,t}\}_{t=2}^{N}$ using equation 3.7
        and then compute the volatility (standard deviation) at time $t=N$, that
        is $\sigma_{i,N} = \sqrt{\sigma^2_{i,N}}$
    2.3 Define a time series $\{S_{i,t}\}_{t=1}^{T}$ to store the context variable simulated values
      2.4 For each trial $j$ $(1 \leq j \leq T)$
          2.4.1 Compute the simulated log return $R^*$
          2.4.2 Compute the j^th simulated value $S_j$ as follows:
                $S_j = V_N^i \exp(R^*)$ where $exp(x) = e^x$ and $V_N^i$ is the value of
                context variable i at time $N$
Output: $S_G = \left(\{S_1\}_{t=1}^{T}, \{S_2\}_{t=1}^{T} \cdots \{S_M\}_{t=1}^{T}\right)$
```

The simulated log R* needs to be computed in such a way that the correlations amongst the context variables returns are maintained. For instance, by using the

Cholesky factorization [10], it can be shown that the formulae to generate the simulated returns for three context variables would be as follows:

$$R_1 = \sigma_1 Z_1,\ R_2 = \sigma_2 \left(\rho_{12} Z_1 + \sqrt{1-\rho_{12}^2}\, Z_2\right),$$

$$R_3 = \sigma_3 \left(\rho_{13} Z_1 + \frac{\rho_{23} - \rho_{12}\rho_{13}}{\sqrt{1-\rho_{12}^2}} Z_2 + \sqrt{1-\rho_{13}^2 - \frac{(\rho_{23} - \rho_{12}\rho_{13})^2}{1-\rho_{12}^2}} Z_3 \right)$$

where σ_i is the volatility (standard deviation) for context variable i, ρ_{ij} is the correlation between context variables i and j and each Z is a generated IID normal value.

In general, generating correlated variates for M context variables can expressed as follows [25]:

$$\begin{aligned} R_1 &= a_{1,1} Z_1 + a_{1,2} Z_2 + a_{1,3} Z_3 + ... + a_{1,n} Z_n \\ R_2 &= a_{2,1} Z_1 + a_{2,2} Z_2 + a_{2,3} Z_3 + ... + a_{2,n} Z_n \\ &\cdots \\ R_M &= a_{M,1} Z_1 + a_{M,2} Z_2 + a_{M,3} Z_3 + ... + a_{M,n} Z_n \end{aligned}$$

or in matrix form

$$R = \begin{pmatrix} R_1 \\ R_2 \\ \cdots \\ R_M \end{pmatrix}, Z = \begin{pmatrix} Z_1 \\ Z_2 \\ \cdots \\ Z_M \end{pmatrix}, A = \begin{pmatrix} a_{1,1} a_{1,2} a_{1,3} \cdots a_{1,n} \\ a_{2,1} a_{2,2} a_{2,3} \cdots a_{2,n} \\ \cdots\cdots\cdots \\ a_{M,1} a_{M,2} a_{M,3} \cdots a_{M,n} \end{pmatrix} \equiv R = AZ$$

The matrix A must satisfy the covariance requirements. By eliminating the random vectors R and Z, we have the following:

$$\begin{aligned} R &= AZ \\ RR^T &= AZ(AZ)^T = AZZ^T A^T \\ E[RR^T] &= E[AZZ^T A^T] = AE[ZZ^T]A^T \\ E[RR^T] &= AIA^T \\ \Sigma &= AA^T \end{aligned}$$

Several methods can be used to solve the set of equations for AT and generate correlated variates, for instance, Cholesky Decomposition [10], Singular Value Decomposition [10] and Return Space Decomposition [1]. The authors in [25], [20, [1] showed that the matrix A can be expressed as follows: matrix A must satisfy the covariance requirements. By eliminating the random vectors R and Z, we have the following:

$$A = \frac{1}{\sqrt{\sum_{i=1}^{n} \alpha^i}} \begin{pmatrix} \alpha^{\frac{1}{2}} r_{1,1} \; \alpha^{\frac{2}{2}} r_{1,2} \; \alpha^{\frac{3}{2}} r_{1,3} \ldots \alpha^{\frac{n}{2}} r_{1,n} \\ \alpha^{\frac{1}{2}} r_{2,1} \; \alpha^{\frac{2}{2}} r_{2,2} \; \alpha^{\frac{3}{2}} r_{2,3} \ldots \alpha^{\frac{n}{2}} r_{2,n} \\ \ldots \; \ldots \; \ldots \; \ldots \\ \alpha^{\frac{1}{2}} r_{M,1} \; \alpha^{\frac{2}{2}} r_{M,2} \; \alpha^{\frac{3}{2}} r_{M,3} \ldots \alpha^{\frac{n}{2}} r_{M,n} \end{pmatrix}$$

where n=N-1 is the number of log returns, $\alpha$is the smoothing factor and $r_{i,j}$ is the return of context variable *i* at time j.

The advantages of using the Return Space Decomposition (RSD) over the Cholesky and SVD factorizations can be summarized as follows [25]:

- In both, Cholesky and SVD factorizations, the decomposed matrix does not easily provide an intuitive understanding of how the future values are generated and the change of a single value of a context risk factor requires a new decomposition. Finally, the Cholesky factorization requires that the correlation matrix be PD (positive definite), and SVD requires PSD (positive semi-definite).
- Volatilities and correlations do not have to be computed when using Return Space Decomposition.

Table 4 describes the modified algorithm for *M* context dependent variables using the Return Space Decomposition:

Table 4. Scenario generation algorithm for multiple dependent context dependent variables using Space Return Decomposition

Input: $\left(\{V_1\}_{t=1}^{N}, \{V_2\}_{t=1}^{N} \ldots \{V_M\}_{t=1}^{N}\right)$
1. Choose the number of scenario trials T and smoothing factor α
2. Compute the α weights vector $lwv = \left(\alpha_1^{\frac{1}{2}}, \alpha_2^{\frac{2}{2}}, \alpha_3^{\frac{3}{2}}, \ldots, \alpha_n^{\frac{n}{2}}\right)^T$ and the $ssq = \frac{1}{\sqrt{\sum_{i=1}^{n} \alpha^i}}$

Table 4. (*continued*)

```
3. For each context risk factor time series $i\ (1 \le i \le M)$
    3.1 Compute the A Matrix, where each
        $a_{ij}, (1 \le i \le M, 1 \le j \le N) = r_{ij} \cdot lmv \cdot ssq$ where $r_{ij}$ is the return of
        context variable $i$ at time $j$.
    3.2 Define a time series $\{S_{i,t}\}_{t=1}^{T}$ to store the context variable simulated values
    3.3 For each j trial $(1 \le j \le T)$
        3.3.1 Compute an N size vector
            $Z = (z_1, z_2, z_3, \ldots, z_n)^T, z_i \sim IID\ N(0,1)$
        3.3.2 For each context variable $k\ (1 \le k \le M)$, compute
            $SLR = (A_k \circ Z) \cdot \sqrt{H}$ where $A_k$ is the row-vector corresponding
            to context variable $k$ and the evaluation horizon $H$.
        3.3.3 Compute the $j^{th}$ simulated value $S_j$ as follows:
            $S_{i,j} = V_N^i \exp(SLR)$ where $exp(x) = e^x$ and $V_N^i$ is the value for
            the context variable $i$ at time $N$ 2.1 Compute the $N$-$1$ returns
            from $\{V_{i,t}\}_{t=1}^{N}$. The result of this calculation is a time series of returns
            of the form $\{r_{i,t}\}_{t=2}^{N}$
Output: $S_G = \left(\{S_1\}_{t=1}^{T}, \{S_2\}_{t=1}^{T} \cdots \{S_M\}_{t=1}^{T}\right)$
```

6 Risk

Due to the dynamic nature of context variables, we will measure the *risk* of each context variable (and for a group of context variables) as the maximum *error* amount that is incurred in the estimation of future values (for each context variable) during an evaluation horizon for a given confidence level. In this framework, to measure the risk of a context variable, the following is assumed:

1. Risk can be measured according to the changes of the context variables *(e.g. location, power resource, etc…).*
2. Without loss of generality, the values of a context variable are always positive. Moreover, the [log] changes in the context variables can be modeled using as a random walk (following a normal distribution).

To calculate the maximum *error* amount for each context variable, we need to create a set of time series $E = \left(\left\{E_{1,t}\right\}_{t=1}^{T}, \left\{E_{2,t}\right\}_{t=1}^{T} \ldots, \left\{E_{i,t}\right\}_{t=1}^{T}, \ldots \left\{E_{M,t}\right\}_{t=1}^{T}\right)$, where $\left\{E_{i,t}\right\}_{t=1}^{T}$ is the estimation error series for context variable *i* and defined as follows:

$$\left\{E_{i,t}\right\}_{t=1}^{T} = S_{i,j} - V_N^i$$

where V_N^i is the value for the context variable *i* at time *N*.

To measure the risk, we use the ***percentile*** function to determine the proportion of values in the time series $\left\{E_{i,t}\right\}_{t=1}^{T}$ that a specific magnitude will not be exceeded by a specific magnitude. This means that, for a context variable, this measure is the maximum error amount (*risk*) that would be incurred in the estimation during the evaluation horizon *H* for a given *confidence level (γ)*. Formally, the p^{th} percentile of the $\left\{E_{i,t}\right\}_{t=1}^{T}$ values is defined as the magnitude that exceeds *p* percent of the values, that is:

$$risk_{i,P} = \left| percentile\left(\left\{E_{i,t}\right\}_{t=1}^{T}, (1-\gamma)\right) \right|$$

Because of the assumption that context values are *normally distributed*, mathematically, the p^{th} percentile (denoted by *α*) of a continuous probability distribution is given by the following formula [13]:

$$p = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\alpha} e^{-x^2/2} dx$$

Finally, assuming that the context variables can be linearly aggregated (Section 2.4) and given that the sum of normal random variables is itself normally distributed [19], then, the overall change time series of a group of *M* context variables is the weighted sum of M underlying returns and can be stated as follows:

$$\left\{E_G\right\}_{t=1}^{T} = \left(\sum_{i=1}^{M} \frac{E_{i,1}}{c_i}, \sum_{i=1}^{M} \frac{E_{i,2}}{c_i}, \cdots, \sum_{i=1}^{M} \frac{E_{i,T}}{c_i} \right)$$

where $E_{i,j}$ is the simulated error for context variable *i* at trial *j*, $c_{i,}$ is the scaling amount for context variable *i*. Then, the overall change (risk) of a group of context variables is

$$risk_{G,P} = \left| percentile\left(\left\{E_G\right\}_{t=1}^{T}, (1-\gamma)\right) \right|$$

7 Security Policies

In our army scenario, a simple security policy can be stated as follows: a Soldier can receive the location of his unit members, provided his *proximity* to the company commander (Captain) is no less than 70 meters. Moreover, the proximity between the Captain and the soldier's unit leader Sergeant) needs to be no less than 50 meters (The Sergeant and the Soldier are within line of sight of each other). The commanding officer requires that the location of the Soldier and the Sergeant be reported every minute and that the *volatility* (standard deviation) of the data not exceed 0.01. Every minute the policy is reevaluated ($H = 1$). The commanding officer (Captain) uses a smoothing factor of 0.95 ($\alpha = 0.95$) and generates 100 Monte Carlo scenarios (*$T = 100$)* before returning a decision. Table 5 shows the description of the example Context Based Security Policy described above.

Table 5. Example of the context based security policy

Name: Unit member location access policy.

Description: A soldier can access location of his unit members according to his current position and proximity of his commanding officer.

Entities:

1) *so*: Soldier, 2) *co*: Commanding Officer, 4) se: Sergeant 3) *LD*: Location data

Context policy parameters:

1) $H = 1$ minute, 2) $T = 200$, $y = \alpha = 0.95$, $N = 10$

Access to *LD* is granted to *so* iff

1. *proximity(so, co)* < 70 meters & proximity(co, se) < 50 meters & proximity (so, se) < 5meters
2. *volatility (proximity(so,co,N))* <= 0.01 & *volatility (proximity(co,se,N)) <=0.01*
3. *error(proximity(so,co,N))* <=0.6

***Functions**:*

- *proximity(x,y): returns the proximity between entities x and y*
- *proximity(x,y,n): returns a time series with the last n proximity values between x and y .*
- *volatility(ts): returns the volatility of the time series ts*
- *error(ts): returns the estimation error through a Monte Carlo simulation for time series ts.*

Security Policy Evaluation. Under the Random Walk - Monte Carlo method framework presented in Sections 3, 4 and 5, a context based security policy requires the following parameters to be defined and evaluated:

1. Smoothing factor (α)
2. The number of scenario trials (T)
3. Evaluation horizon (H)
4. Time series frequency (F)
5. Number of observations required in a context variable time series (N)
6. The confidence level (γ) to be used for risk evaluation
7. For each context variable in the policy, a risk (error) tolerance is defined.

The following steps are used to evaluate the policy:

1. A soldier so sends a request to his commanding officer co for the location of his unit members.

2. The co retrieves the locally stored security policy, reads it and determines which context variables (i.e. proximity) are necessary for the evaluation of the policy
3. The co sends a request asking both the so and the se for their proximity information. Such a request contains:
   a. The context variable to be gathered (e.g. proximity)
   b. The number of past observations (N) in the context variable time series.
4. Upon receiving the requests both the so and se return the required time series information to the co.
5. The co runs the modified Monte Carlo algorithm (Table 3) to calculate the 1-minute volatility. It then calculates the risk of the context variables (proximity) with the specified confidence level (y) using the percentile function.
6. Once the data is computed the co determines if access to the data LD can be granted for the next 1 minute (H = 1) before the policy needs to be revaluated.

8 Related Work

There are several research efforts in developing security systems that use context variables to either solely render a security decision or expand Role Based Access Control (RBAC) implementations and policies. For instance, [12] defines a security system that only uses context data to arrive at a security decision while keeping the anonymity of the accessing users. However the system does not use any context data history to render a more accurate security decision. In [21] a theoretical model to formalize and represent situation-based security policies using context graphs is proposed. However, no provision for the use of historical context data in the enforcement of security is given. The system in [17] defines a programming interface to handle inexact information from multiple sources while checking the honesty of the input context data; however it assumes that security of the data must be handled by the application using the interface.

The grammar explained in [32] allows the expansion of traditional RBAC policies with location-dependent data. However, the grammar does not include predicates to validate or use history of context data. The model in [36] expands the role based access control (RBAC) paradigm while a user agent adjusts role permissions based on context data. However, the system does not provide for a way to invalidate erroneous context claims, which may cause information leaks.

There are several systems that use the concept of trust and risk to deliver a security decision. For instance, the model in [7] characterizes an outcome-based approach to allow trust reasoning and calculate cost data per-outcome basis, instead of relying on a simple risk metrics value. Trust is then computed using current evidence along previous observations and recommendations. However, the system has not been fully validated for pervasive mobile environments. In [16] a model that takes into account the trust dynamics of past experiences, including intentions or beliefs is presented. The model requires each entity to count its own positive and negative evidences about others. However, the trust scheme does not use any context information to refine trust values.

The preliminary work done in [14] attempts to model the relationship between risk and trust in order to derive a computational model that integrates the two concepts. It uses the probability of success in a transaction between two entities. However, the system does not formally describe or model the risk in terms of context information. The research described in [3] uses Kalman-Filter equations to calculate trust between a client and a service provider based on the discrepancies found amid the service advertised quality values and the client's measurements. However, since the model is outcome based, it does not use context information to refine the trust measures. The research described in [29] illustrates a mechanism to compute trust between entities based on their own direct interaction experiences as well as recommendations. However, trust calculations do not take into account the entities' context information.

The work described in [37] extends the RBAC model with spatial and location-based information where spatial entities are used to model objects and user positions. In this system, security roles are activated based on the positional information from a user. Although very useful security system for geographical information systems (GIS), the system cannot be easily adapted to small mobile devices or implemented in a mobile ad-hoc network (MANET) environment.

The preliminary work in [26] introduces the random walk presented in Section 2.6 and the use of the percentile function to calculate the risk of a context variable in a security policy. However, this work does not take into account the correlations amongst different context variables in a policy.

9 Conclusions and Future Work

In this paper we presented a framework to model context variables that takes into account the relationships amongst them for secure collaborations. In this framework, security decisions are reached by measuring the risk of each context variable according to its value changes. To reach such a decision, we introduced a novel method called space return decomposition to evaluate a context-based security policy without calculating correlations and volatilities. In addition, we presented an example of a real world policy where our framework can be applied.

For future work, more investigation is necessary to include or derive a trust measure from the context data for the entity (user) requesting access permission. Finally, analysis is necessary to determine the appropriate amount of history to keep (or for bootstrapping) before a reliable computation can be made.

References

1. Benson, P., Zangari, P.: A General Approach to Calculating VaR without Volatilities and Correlations. JPMorgan/Reuters RiskMetrics Monitor (Second Quarter 1997)
2. Boeing. "ScanEagle." The Boeing Corporation, All rights reserved, http://www.boeing.com/defense-space/military/scaneagle/index.html (accessed, November 2007)
3. Capra, L., Musolesi, M.: Autonomic Trust Prediction for Pervasive Systems. In: International Conference on Advanced Information Networking and Applications, vol. 2, pp. 481–488 (2006)

4. Castelli, G., et al.: A Simple Model and Infrastructure for Context-Aware Browsing of the World. In: IEEE International Conference on Pervasive Computing and Communications, pp. 229–238 (March 2007)
5. Chong, C.-Y., Kumar, S.P.: Sensor Networks: Evolution, Opportunities and Challenges. Proceedings of the IEEE 91(8), 1247–1256 (2003)
6. Chou, Y.L.: Statistical Analysis. Copyright © 1975 Holt International (1975) ISBN 0030894220
7. Dimmock, N., Belokosztolszki, A., Eyers, D.: Using Trust and Risk in Role-Based Access Control Policies. In: ACM Symposium on Access Control Models and Technologies, pp. 156–162 (June 2004)
8. Dimmock, N., Bacon, J., Ingram, D., Moody, K.: Risk Models for Trust Based Access Controls (TBAC). In: Herrmann, P., Issarny, V., Shiu, S.C.K. (eds.) iTrust 2005. LNCS, vol. 3477, pp. 364–371. Springer, Heidelberg (2005)
9. FAS Military Network Analysis. "Land Warrior." Federation of American Scientists, `http://www.fas.org/man/dod-101/sys/land/land-warrior.htm` (accessed, January 2007)
10. Golub, G.H., Van Loan, C.F.: Matrix Computations, Section 4.2, 3rd edn., pp. 140–152. Johns Hopkins University Press (1996) ISBN 0-8018-5414-8
11. Hamilton, J.D.: Time Series Analysis. Copyright © 1994. Princeton University Press, Princeton (1994)
12. Hulsebosch, R., Salden, A., Bargh, M., Ebben, P., Reitsma, J.: Context Sensitive Access Control. In: ACM Symposium on Access control models and technologies, pp. 111–119 (June 2005)
13. JPMorgan/Reuters. RiskMetrics^{TM} – Technical Document 4th Edition. RiskMetrics Group, New York (1996), `http://www.riskmetrics.com`
14. Jøsang, A., Presti, S.L.: Analyzing the Relationship between Risk and Trust. In: Jensen, C., Poslad, S., Dimitrakos, T. (eds.) iTrust 2004. LNCS, vol. 2995, pp. 135–145. Springer, Heidelberg (2004)
15. Kim, J., Malz, A.M., Mina, J.: LongRun. Technical Document, RiskMetrics Group (1999)
16. Korpipää, P., Mäntyjärvi, J., et al.: Managing Context Information in Mobile Devices. IEEE Pervasive Computing 2(3), 42–51 (2003)
17. Lin, C., Varadharajan, V., Wang, Y., Pruthi, V.: Trust Enhanced Security for Mobile Agents. In: IEEE International Conference on Ecommerce Technology, pp. 231–238 (July 2005)
18. Litterman, B.: Modern Investment Management: An Equilibrium Approach (Hardcover). In: Copyright © 2003, Goldman Sachs Inc., John Wiley & Sons Inc., Chichester (2003)
19. Mendenhall, W., Scheaffer, R.L., Wackerly, D.D.: Mathematical Statistics with Applications. Copyright © 1986, 3rd edn. PWS Publishers (1996) ISBN 0-87150-939-3
20. Mina, J., Xiao, J.Y.: Return to RiskMetrics: The Evolution of a Standard. RiskMetrics Group (April 2001)
21. Mostefaoui, G.K., Brezillon, P.: Modeling Context-Based Security Policies with Contextual Graphs. In: IEEE Annual Conference on Pervasive Computing and Communications Workshop, pp. 28–33 (March 2004)
22. Neumann, G., Strembeck, M.: Access Control: Design and Implementation of a Flexible RBAC-Service in an Object-Oriented Scripting Language. In: ACM Conference on Computer and Communications Security, pp. 58–67 (November 2001)
23. Park, J.S., Costello, K.P., Neven, T.M., Diosomito, J.A.: Access management for distributed systems: A composite RBAC approach for large, complex organizations. In: ACM Symposium on Access Control Models and Technologies, pp. 163–172 (June 2004)

24. riskglossary.com “Return.” Copyright © Contingency Analysis, 1996 - current (1996), http://www.riskglossary.com/link/return.htm
(accessed, December 2006)
25. RiskMetrics Group. “Risk University.” RiskMetrics Group, http://riskuniversity.org/ (accessed, May 2006)
26. Sanchez, C., Gruenwald, L., Sanchez, M.: A Monte Carlos Framework to Validate Context-Based Security Policies in Pervasive and Mobile Environment. In: ACM SIGMOD International Conference on Data Engineering for Wireless and Mobile Access (MobiDE) (June 2007)
27. Rose, C., Smith, M.D.: The Multivariate Normal Distribution. In: Section 6.4 in Mathematical Statistics with Mathematica, pp. 216–235. Springer, New York (2002)
28. Swift, M., Hopkins, A., Brundrett, P., Dyke, C.V., Garg, P., Chan, S., Goertzel, M., Jensenworth, G.: Improving the granularity of access control for Windows 2000. ACM Transactions on Information and System Security 5(4), 398–437 (2002)
29. Teacy, W.T.L., Patel, J., Jennings, N.R., Luck, M.: TRAVOS: Trust and Reputation in the Context of Inaccurate Information Sources. Journal of Autonomous Agents and Multi-Agent Systems 12(2), 183–198 (2006)
30. Uryasev, S.: Introduction to the Theory of Probabilistic Functions and Percentiles (Value-at-Risk). In: Probabilistic Constrained Optimization: Methodology and Applications, pp. 1–25. Kluwer Academic Publishers, Dordrecht (2000)
31. Want, R.: Introduction to RFID Technology. IEEE Pervasive Computing 5(1), 25–33 (2006)
32. Wedde, H.F., Lishka, M.: Role-Based Access Control in Ambient and Remote Space. In: ACM Symposium on Access Control Models and Technologies, pp. 21–30 (June 2004)
33. Weisstein, E.W.: Normal Distribution. From MathWorld–A Wolfram Web Resource, http://mathworld.wolfram.com/NormalDistribution.html (accessed, August 2006)
34. Weisstein, E.W.: Random Walk–1-Dimensional. From MathWorld–A Wolfram Web Resource,
http://mathworld.wolfram.com/RandomWalk1-Dimensional.html
(accessed, August 2006)
35. Chou, Y.-L.: Moving Average. Statistical Analysis for Business and Economics. Holt Rinehart and Winston Publishers (January 1989) ISBN 978-0444013019
36. Zhang, G., Parashar, M.: Context-aware Dynamic Access Control for Pervasive Applications. In: Communication Networks and Distributed Systems Modeling and Simulation Conference, 2004 (January 2004)
37. Bertino, E., Damián, M., Catania, B., Perlasca, P.: GEO-RBAC: a spatially aware RBAC. ACM Transactions on Information Systems and Security 10(1) (February 2007)