Journal of Computer Science and Information Technology
December 2014, Vol. 2, No. 3 & 4, pp. 83-105
ISSN: 2334-2366 (Print), 2334-2374 (Online)

Published by American Research Institute for Policy Development
DOI: 10.15640/jcsit.v2n3-4a5
URL: http://dx.doi.org/10.15640/jcsit.v2n3-4a5

Context Dependent Threat-Based Access Control System

Adejimi Alaba Olusesi[1] & Sodiya Adesina Simeon[2]

Abstract

As technology advances, protecting network resources against unauthorized access and misuse of privileges became more necessary and the present access control system seems not efficient enough to solve the problems. This work presents a Context Dependent Threat-Based Access Control (CDTAC) system for correcting these problems. The CDTAC is an adaptive access control system that combines both static and dynamic information (contextual parameters) of users to adjust access control decisions based on environmental threat factors for efficient decision making. The design adopts relative probability in the estimation of the threat level of the contextual parameters. An improved Multifactor Evaluation Method was also used to estimate the associated risk attached to the contextual parameters in accordance with the information security objectives. The weighted arithmetic mean was then employed to evaluate the associated risk attached with users' requests based on the impact of the threat. In making final access decisions, the risk threshold values (i.e. $0 \leq \varphi_1 < \varphi_2 \leq 1$) were set in relation to the level of sensitivity of the resources. The evaluation result showed an acceptable security index of 0.18 and 99.1% compliance level of CDTAC.

Keywords: contextual, threat, risk threshold, access control

1. Introduction

Many organizations now depend on computer-based systems for their daily activities. Large amount of information are processed, stored and managed on these systems.

[1] Department of Computer Science, Federal University of Agriculture, Abeokuta, Ogun State, Nigeria. Email: a_olasumbo@yahoo.com, Phone: +234-8062433677
[2] Department of Computer Science, Federal University of Agriculture, Abeokuta, Ogun State, Nigeria. Email: sinaronke@yahoo.co.uk, Phone: +234-8034551851

Consequently, protecting these sensitive information is an inevitable task and controlling access to resources on ICT infrastructure require different protection requirements based on their importance. The greatest threat to computer systems and their information comes from humans; through actions that are either deliberate or ignorant. When the action is deliberate, some motivation or goals are generally behind it. For instance, the goal could be to disrupt normal business operations, thereby denying data availability and production.

Sodiya and Onashoga (2009) stated that access control is concerned with limiting the activity of legitimate users who have been successfully authenticated thereby ensuring that every access to a system and its resources is controlled and only those access that are authorized can take place. In some cases, authorization may mirror the structure of the organization, while in others it may be based on the sensitivity level of various documents and the clearance level of the user accessing those documents (NIST, 2006). It could also be seen as the configuration of those controls to protect information and information systems from unauthorized access, use, disclosure, disruption, modification or destruction in order to provide confidentiality, integrity and availability (Gleneesha, 2010). There are three basic components in an access control system: the subjects, the targets and the rules which specify the ways in which the subjects can access the targets. The entity that requests access to a resource is called the **subject** of the access; it is an active entity because it initiates the access request while the resource a subject attempts to access is called the **target / object** of the access. Security-relevant context consists of the set of contextual attributes that can be used to characterize the situation of an entity, whose value affects the choice of the most appropriate controls (measures).

Information security has three separate but interrelated objectives and these are the components through which information can be compromised: these are;

(1) C*onfidentiality* (or *secrecy),* the prevention of unauthorized disclosure of information. This can be the result of poor security measures or information leaks by personnel. An example of poor security measures would be to allow anonymous access to sensitive information.
(2) I*ntegrity,* The prevention of erroneous modification of information. Authorized users are probably the biggest cause of errors, omissions and the alteration of data. Storing incorrect data within the system can be as bad as losing data.

Malicious attackers also can modify, delete, or corrupt information that is vital to the correct operation of business functions.

(3) A*vailability,* The prevention of unauthorized withholding of information or resources. This does not apply just to personnel withholding information. Information should be as freely available as possible to authorized users.

Traditional approaches to security were developed when users were typically computing in a static, stationary environment, and therefore base security-related decisions on static attributes such as identity and role (Gleneesha et al., 2011). Contextual access control policies provide the means to handle complex security system requirements in a flexible and dynamic manner. Context reflects the combination of quantifiable data that may be relevant to an access control decisions. The definition includes (but is not limited to) the user's spatiotemporal setting, his access request history, the device used to make the request, the trust placed in the user by the organization, the time of access, the frequency of access requests and the presence of an emergency situation.

A lot of problem is encountered when making decision and because of the unpredictable nature of the environment, there is need to utilize context to dynamically adjust users' permissions that are commensurate with its risk level based on the user's current contextual attributes. Risk is the potential harm that may arise from some present processes or from some future events. Risk management is the process of identifying vulnerabilities and threats to the information resources used by an organization in achieving business objectives, and deciding what counter measures, if any, to take in reducing risk to an acceptable level, based on the value of the information resource to the organization. It is a function of the likelihood of threats to the system being realized and the resulting impact (CISA, 2006).

In the context of this paper, an efficient access control systems that adequately protect network resources against unauthorized access is developed. The proposed system is also designed to handle internal abuses by controlling users' privileges. Environmental threats associated with the contextual parameters are considered and measured through some defined policies based on the users' domain.

The risk attached to any possible action initiated by the user is measured by taking into consideration the different components through which information can be compromised (i.e. confidentiality, availability and integrity), and the sensitivity level of each resource is also considered before final access is given to any resource within an organization.

The rest of this paper is organized as follows; section 2 is the reviewed of the previous related works about role-based access control system, context-based access control and threat/risk-based access control systems. Section 3 presents the architecture of the method employed for the Context Dependent Threat-Based Access Control (CDTAC) system. Section 4 discusses the implementation and performance evaluation. Section 5 presents the future work and conclusion.

2.0 Related Works

Several models of access control have been developed for improving computer system security; their corresponding access control mechanisms, the concrete implementations of those access control models and underlying infrastructure components involve varying degrees of complexity. In many cases, the newer, more complicated models arose not from deficiencies in the security that earlier models provide, but from the need for new models to address changes in organizational structures, technologies, organizational needs, technical capabilities, and/or organizational relationships.

2.1 Role-Based Access Control

Role-based access control (RBAC) is a method of regulating access to computer or network resources based on the roles of individual users within an enterprise. Roles are defined according to job competency, authority, and responsibility within the enterprise. In RBAC, Users can be made members of roles as determined by their responsibilities and qualifications and can be easily reassigned from one role to another without modifying the underlying access structure. Roles can be granted new permissions as new applications and actions are incorporated, and permissions can be revoked from roles as needed. Recently, RBAC was found to be the most attractive solution for providing security features in different distributed computing infrastructure. All RBAC models share the same basic structure of subject, role and privilege.

A Typical Role-Based Access Control (RBAC) systems were presented in Ferraiol (2003). They presented architecture for ensuring separation of duties in order to control access to computer resources. The problem with RBAC is that it is difficult in some cases to encapsulate all permissions to perform a job function. In fact, role engineering has turned out to be a difficult task (NISTIR, 2006).

It cannot capture any security relevant information from its environment due to the subject-centric nature of its roles.

Thus, in the new ubiquitous environment, RBAC is not sufficient for adequate security.

2.2 Context-Based Access Control

Contextual access control policies provide the means to handle complex security system requirements in a flexible and dynamic manner. In the dynamic computing environment of mobile workers, users may: 1) use a variety of mobile computing devices with varying configurations; 2) connect over various networks; and 3) be in varying physical settings when requesting access to remote resources (Gleneesha et al., 2011). To achieve effective security in this dynamic computing environment, security decisions must consider users' context (e.g., co-location, network characteristics, and device characteristics), which can change frequently and rapidly. Context is an important factor in making access control decision when a subject wants to access a sensitive object. Al-Rwais et al. (2010) defined context as "any information that can be used to characterize the situation of an entity". Examples of context include user's identity, user's location, user's vital signs, surrounding environment, network bandwidth, available processing power, etc. In a full context based environment the access decision may be based on static information such as: user identity (user name or derived from badge, token, etc.); authentication data (such as password, certificate or biometric information); group and role membership; the day and time the service is allowed to be accessed. This information can be combined with dynamic information, e.g. client application used to access the service; minimum patch levels installed, the location of the user during the request (office or remote); network security level used (secured wireless, LAN, VPN); the day and time the service is accessed; other roles of the user that might conflict with the applicable role (dynamic separation of duty); other details about the request (number of copies for printing, viewing of restricted data).

Young et al. (2005) proposed a context-aware AC model which considers location, time, and system resources as AC constraints. The role is activated only if all the constraints are satisfied. The model has failed to consider the potential composite effects of, or the correlations between, these context attributes.

The proposed contextual RBAC model classifies the patients' records based on their sensitivity levels, and an AC decision is made based upon the sensitivity level of the data being requested. The work, however, does not show how to adjust AC decisions in adaptation to the requesters' dynamic changes of the contextual information.

Nguyen (2009) tried to address the need for evaluating the effect of multiple contextual attributes on an authorization decision coherently. The model introduces the notion of risk-aware AC. The context information is used as the input to a risk assessment process to compute a risk value that is then fed into the authorization decision engine. However, the scope of the risk assessment is quite broad covering confidentiality, integrity and authentication, so the delay incurred in the risk value calculation may be quite large, which may adversely affect the performance of the underlying AC system.

2.3 Threat / Risk-Based Access Control

Risk is the potential harm that may arise from some present processes or from some future events. Risk management is the process of identifying vulnerabilities and threats to the information resources used by an organization in achieving business objectives, and deciding what countermeasures, if any, to take in reducing risk to an acceptable level, based on the value of the information resource to the organization. It is a function of the likelihood of threats to the system being realized and the resulting impact (CISA, 2006).

(Sven, 1994) mentioned in his work that risk assessment is an effective tool to be used in decision making and is the determination of quantitative or qualitative value of risk related to a concrete situation and a recognized threat (also called hazard). There are two methods of risk assessment in information security field, qualitative and quantitative. Quantitative risk assessment requires calculations of two components of risk: *R*, the magnitude of the potential loss *L*, and the probability *p*, that the loss will occur. It can be expressed as:

$$R_i = L_i P(L_i) \quad \text{----------(1)}$$

Gleneesha et al. (2011) presented an approach that used generalized annotated programs (GAPs) to practically incorporate context into security services with a focus of access control. He determined the overall threat level of the user current situation by combining a logic-program that encoded an administrator's policies with the user's context. In using the paradigm, he was able to associate real-numbered value with contextual attribute and can directly leverage results to correctly determine the threat level entailed by a user's context. The work did not really look into getting the precise risk value posed by the threat and the designed GAPs system is not evaluated in a test environment.

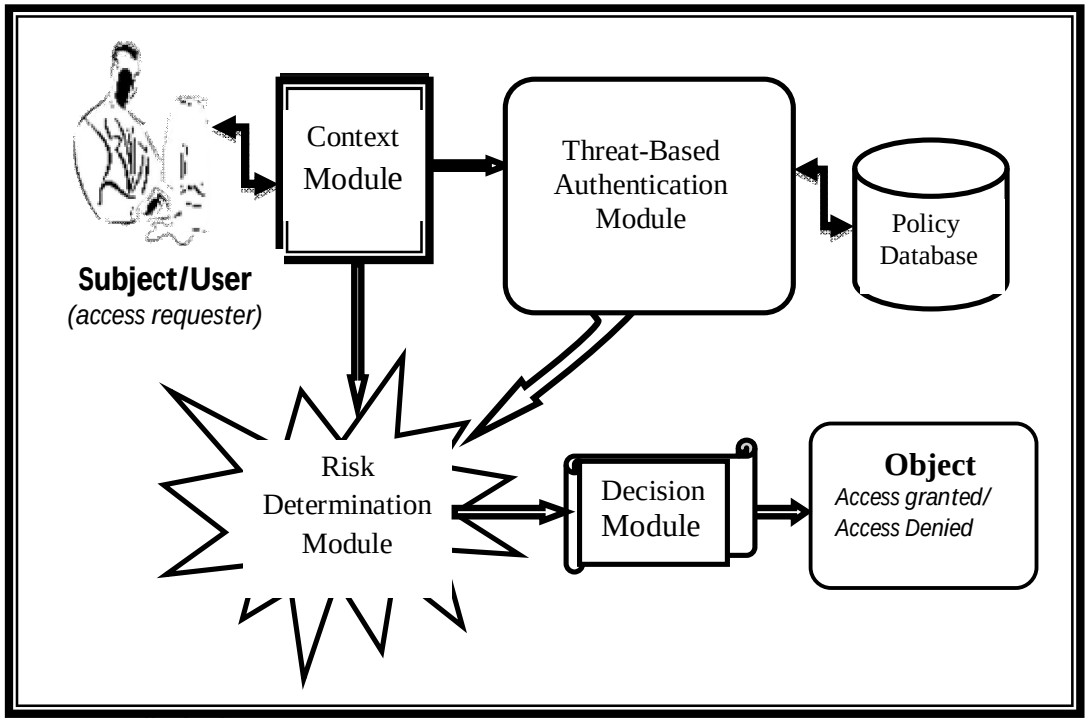
Sodiya et al. (2009) presented an access control scheme that adopted the techniques of Role-Based Access Control (RBAC), Purpose-Based Access Control (PBAC), Time-Based Access Control (TBAC) and History-Based Access Control (HBAC) as components to form an integrated Components-based Access Control Architecture (CACA). In CACA, an Access Control Score (ACS) is computed from the combined access control techniques. CACA also combines ACS with the sensitivity nature of system resources before a level of access is granted. CACA did not consider the threat that may arise from the environment and the risk attached to granting access to sensitive resource.

3.0 Design Methodology

3.1 Architecture of Context Dependent Threat-Based Access Control (CDTAC) System

Subject centric (traditional) access control systems are very useful access control models but due to the distributed and heterogeneous nature of organizations, they are no longer sufficient. With the rapid advancement in technologies today, organizational resources are widely distributed. Under these circumstances, an extension of subject centric model is necessary in order to properly manage the organizational resources in multi domain environment keeping in mind the information security objective of confidentiality, integrity and availability. This is proposed as CDTAC, a dynamic access control model that allows security administrators to define context oriented access control policies enriched with the notion of risk and user permission taking into consideration the threat in the environment.

The proposed Context Dependent Threat-Based Access Control (CDTAC) mechanism consists of four (4) modules. The main modules are the threat-based authentication module, the risk determination module and the decision module. The context module is the fourth module that supports other modules. Some information from environment was utilized and processed in a novel risk assessment model based on a modified multifactor evaluation process which then take into consideration the three important factors of security: availability, integrity and confidentiality, based on users' domain. Moreover, a predefined threshold value was assigned and threshold was set between 0 and 1 to categorize the resource/object based on its sensitivity level. Figure 1 shows the model for the proposed CDTAC system.



3.2 Mechanism of the Context Dependent Threat-Based Access Control (CDTAC) System

There are a number of factors that can increase the risk of unauthorized access; each of these factors becomes a threat to the system in consideration when wrongfully declared. In this work, the focus is on the role of the user, the access locations, the time of access, and the frequency of access (i.e. user's access periods). These factors are known as contextual parameters.

Risk is a function of the likelihood of threats to the system being realized and the resulting impact. To optimally implement an access control security system, it is essential to know the level of threat present in the environment being secured. The level of harm that can be incurred from each contextual attribute called the Threat-Level (T_L) is determined for each factor and the relative probability of the determined Threat-Level is calculated. For each of the contextual attribute, the relative probability (Rel.(Prob.)) of each attribute is determined; then an enforcement algorithm is now deployed to enforce the access policy on the determined threat level of each factor so as to get the overall threat level of the context captured by the system from the user. Figure 2 shows the procedure on how each entity in the proposed framework takes input from each other to arrive at a better decision.

Step 1. Get request/action from the user through the context module
Step 2. System captures context parameters
Step 3. Forward the context parameters to threat-based authentication module
Step 4. Threat-based authentication module enforces the access policy/constrains on the context parameters to get the threat-level of the context parameters using a convectional enforcement algorithm.
Step 5. Threat = Level of harm that can be incurred from each contextual attribute
If(threat = high) then,
deny access
Else if threat('mild' .or. 'low')then,
forward request to the risk determination module
End if.
Step 6. Risk determination module calculates the risk value of the action requested in terms of availability, integrity and confidentiality using an improved Multifactor Evaluation Process (MFEP).
Step 7. Calculate the weighted arithmetic mean (risk value) of its risk value of availability, confidentiality and integrity. Forward the evaluated risk value (RV) to the decision module.
Step 8. Group objects/resources into different categories according to their importance or sensitivity and activate permissions based on user-role permissions.
Step 9. Decision module collects the forwarded risk value (RV), compare the RV result with The threshold access level and return decisions.

Figure 2: Algorithm for Contextual Threat-Based Access Control Mechanism

3.3 Context Module

A request from subject/user to perform an action is submitted to the context module. The requester (subject) provides the credentials such as identification number, password, etc. to the context module and the module captures other context parameters such as role, location, access time and access frequency from the database. The module forwards the context to threat-based authentication module for access processing.

Let U be the set of subject/user
U = {set of subject}
Subject (user) submit request and credentials (static), these are:
i. User name and Password
System captures other contexts from the database, these are:
i. Role
ii. Location
iii. Time of access
iv. Frequency of access (i.e access periods)
These parameters are forwarded to threat-based authentication module.

3.4 Threat-based Authentication Module

The threat-based authentication module has two parts: Enforcement Point (EP) and Policy Database (PD). Access control policies defined resides within the PD.

Enforcement Point (EP) receives access requests from the subject/user through the context module, queries the Policy Database for access control policies, collects other parameters and enforces the access policy on the subject's contextual information using a conventional access control enforcement algorithm.

Enforcement is a way or means of deploying the policy over the system i.e configuring the security components mechanism so that the system behaviour is finally the one specified by the policy. Enforcement algorithm is not a standard algorithm to be quoted; it solely depends on the policy or rules defined by the organization before certain access could be granted on some resources.

3.5 Policy definition and Enforcement

These are the access control policies defined for the CDTAC system.

Policy Definitions

(a) U = $\{u_1,u_2,\ldots,u_n\}$ => set of all registered users/subjects
(b) R = $\{r_1,r_2,\ldots,r_{j}\}$ => set of j user/subject roles in the system
(c) L = $\{l_1,l_2,\ldots,l_k\}$ => set of k logical locations in the system
(d) T = $\{t_1,t_2,\ldots,t_p\}$ => set of p subject/user time of access in the system
(e) AF = $\{af_1, af_2,\ldots, af_h\}$ => set of h access frequencies (i.e. number of previous access) in the system.
(f) RT = $\{rt_1,rt_2,\ldots, rt_q\}$ => set of q times assigned for each role (role-time)
(g) PRM = $\{prm_1, prm_2,\ldots, prm_y\}$ => set of y permissions in the system
$\forall$ *user/subject* $\exists$ *permission given to each role*, then
(h) RPM = $\{(r.prm)_1, (r.prm)_2, \ldots, (r.prm)_t\}$ => set of t user-role permissions
(i) OBJ = $\{obj_1,obj_2,\ldots, obj_m\}$ => set of m objects/resources in the system
$\forall$ i $\in$ N (for all i which is in N)

Then, for each contextual parameter defined (i.e. role, location, time, and access frequency), Threat-Level (T_L) will be determined.

3.6 Determination of Treat-Level(s)

The various threat computations are:

(a) Role Threat-Level

Let the total number of roles defined = N
Let the total number of previous threat by subject/user i = PT_i
For threat = 1 to M

$$\text{Average Threat} = \frac{\text{Sum of threats in the system, } (\sum M)}{\text{Number of roles defined, (N)}} \quad \ldots\ldots\ldots\ldots(2)$$

$$\text{Average Threat per subject} = \frac{\sum M}{N}$$

If (PT_i < Average Threat)
Then,
Threat-Level (T_L) = 0
Else if (PT_i = Average Threat)

Threat-Level (T_L) = 1
Else if (PT_i > Average Threat)
Threat-Level (T_L) = 2
End if.

(b) Location Threat-Level
L = $\{l_1, l_2, \ldots, l_k\}$ => set of access location
If the location is within the Network
Then,
Threat-Level (T_L) = 0
Else
Threat-Level (T_L) = 1

(c) Access Time Threat-Level
For every subject (u) ∃ $role_i$ and
For every role (R_i) ∃ periods assigned for the role (i.e role period)
If (Subject's access time (T_i) ∈ role period (rp_i))

Then
Threat-Level (T_L) = 0 (Normal)
Else if (Access time (T_i) ≠ role period (rp_i)) or (Access time ∩ role period = ∅),
Then
Threat-Level (T_L) = 1(Abnormal)
End if.
T_i = [Normal, Abnormal]

(d) Access Frequency Threat-Level
Let PA be the previous access/transactions by the subject/user i, (PA_i)
Let TSA be the total number of successful access/transaction in the system
Let TSD be the total number of subjects defined

$$\text{Average Access} = \frac{\text{Total number of successful access in the system, (TSA)}}{\text{Total number of subjects defined, (TSD)}} \quad \ldots(3)$$

Average Access per subject = TSA / TSD
If (PA_i > Average Access)

Then,
Threat-Level (T_L) = 0
Else
If (PA_i = Average Access),
Then,

Threat-Level (T_L) = 1
Else
If (PA_i < Average Access),
Then,
Threat-Level (T_L) = 2
End if.

3.6.1 Access Policy (Enforcement Algorithm)

This algorithm enforces the access policy on the context parameters to get the threat level of the context captured by the system.

Set context parameters
$\forall i \in \{r, l, t, af\}$ *where*
l =location
r = role
t = access time
af = access frequency
for U : {set of all users}

For all transaction involving u, i
Begin (enforcement*)*
For every $i \in \{l, r, t, af\}$
Extract T_L *from PD for each attribute(y)*
where $y \in i$.
//Find the Relative Probability of each attribute Threat-Level//

$$\text{Rel. Prob.}(T_L) = \frac{T_{L(i)}}{\sum(T_{L(i)})}$$

$\forall T_L \in \{l, r, t, n\}$
begin
if Rel. Prob. $\sum (T_L) \leq 0.5$
then
access = 1 (where 1 is a true Boolean function)
forward k (k = user request).
else
access = 0 (halt user).
end if
end.

- The EP will release the access grant based on the Threat-Level of the contextual information derived from the enforcement algorithm and forward the result to the risk determination module if the request is granted; if the Threat-Level of the context parameter is too high the EP will deny the requester without further processing.
- Enforcement Point forwards the user's request to risk determination module to calculate the risk value.

3.7Risk Determination Module

Risk is the potential harm that may arise from a present situation, which is often evaluated based on the probability of the threat occurrence and the potential impact. The Risk determination module calculates the risk value of the action requested in terms of the information security objective of availability, integrity and confidentiality using an amended Multifactor Evaluation Process which is adapted for the problem**.**

Formal Statement of Multifactor Evaluation Process (MFEP)

In reality, we have many decision making problems that need to consider many factors. MFEP deals with these problems with a quantitative approach in cases where all of the important criteria can be given appropriate numerical weights and each alternative can be evaluated quantitatively in terms of these criteria.

Given a set of feasible decisions X, the tuple of local characteristics (factors) (k_1, ... k_n) of the decisions x Є X, and the mapping $k_i : x \rightarrow E^1$, i = 1, n. It is required to construct a multifactor estimate of the decision x Є X,

$K = F[k_1(x) ..... k_n(x)]$. ……..(4)

This establishes an order relation on the modified MFP.

The Multifactor Evaluation Process had been in existence for long. Based on MFEP method, a risk determination module was proposed for the system to arrive at decision. The framework consists of five steps as follows.

-Step 1: identify the action requested
-Step 2: calculate the risk value (RV) attached to each context

-Step 3: Assign weight for each factor availability, integrity, confidentiality to the service.
-Step 4: get the risk value of the action requested in term of availability, integrity and confidentiality from the calculated RV of the context parameter and the assigned weight vectors.
-Step 5: find the mean of the RV of the action requested in term of availability, integrity and confidentiality.

Mathematically, the risk value is expressed as:

Risk Value = (Impact of the threat) * (Probability that the threat will occur)
$RV_{(u)}$ = (impact of the threat) * (1- relative probability of the context parameter)

Relative probability of the context parameter = $\frac{T_{Li}}{\sum T_{Li}}$

$$\mathbf{RV_{(u)}} = \sum T_{L(i)} * \left(1 - \frac{T_{Li}}{\sum T_{Li}}\right) \quad \text{.....(5)}$$

For every $i \in \{I, r, t, af\}$ *and* $\forall\ u \in U$.

Risk of Action based on AIC (Availability, Integrity and Confidentiality)

Since this work is centered on the three components of information security objectives, then the risk value of the action in terms of availability, integrity and confidentiality one after the other will be calculated based on the weight vectors assigned to each according to their importance. However, the sum of the weight vectors should be less than or equal to one i.e $\sum w_i \leq 1$.

For availability:

$$RV_{A(a_i)} = RV_{(u)} \cdot w_a \quad \text{.....(6)}$$

For integrity:

$$RV_{I(a_i)} = RV_{(u)} \cdot w_i \quad \text{.....(7)}$$

For confidentiality:

$$RV_{C(a_i)} = RV_{(u)} \cdot w_c \quad \text{.....(8)}$$

in which $i, w_i \in N$ and i = a, i, c.

Evaluating the risk value of the action requested, we find the weighted arithmetic mean of its risk value of availability, confidentiality and integrity. Precisely, it can be calculated as:

$$RV(a_i) = \left(\frac{RV_A(a_i) + RV_I(a_i) + RV_C(a_i)}{w_a + w_I + w_c} \right) \Big/ N \quad \text{.....(9)}$$

where $w_i \in N$, i =a, I, c and they can be adjusted to a suitable value depending on the domain and N is the number of the context parameters used.

3.8 Object Permission

OBJ = $\{obj_1, obj_2, \ldots, obj_m\}$ =>set of m objects/resources in the system
PRM = $\{prm_1, prm_2, \ldots, prm_y\}$ =>set of y permissions in the system
$\forall$ *user/subject* $\exists$ *permission given to each role*, and
$\forall$ Role (r) $\exists$ obj (prm)
R_i ==>$obj_i.prm_i$
For every object/resource,
$obj_i \in \{Role_i.Prm_i\}$
Objects/Resources are categorized or grouped in terms of thei importance/sensitivity.
Category one (Cat.1) object ➔ Highly Sensitive
Category two (Cat.2) object ➔ Sensitive
Category three (Cat.3) object ➔ Not Sensitive

3.9 Decision Module

Comparation of the Threshold Value (φ_1, φ_2) with the Evaluated Risk Value

The context parameters create different Threat-Level for the system. The effect of the resource to risk value depends on properties of resource and we should have some predefined threshold. Threshold denotes a bearable limit assigned by the system administrator either to further process the request or reject the request based on evaluated risk. The threshold is assigned φ_1 and φ_2 and the limit is set between 0 to 1, (i.e. $0 \leq \varphi_1 < \varphi_2 \leq 1$). The decision stage will link the user to the requested resource based on the resource category. Assuming $\varphi_1 = 0.25$ and $\varphi_2 = 0.50$.

Decision

//* Level 1 access: *Category one object/resource*//*

If (RV < φ_1), then
Access granted iff (obj/resource ∈ user-role permission)
=> Access = 1 iff (obj ∈ r_i . p_i)
Else level 2 access.
End if.

//* **Level 2 access: *Category* two *object/resource*//***
If ($\varphi_1 \leq RV \leq \varphi_2$), Then
Access granted iff (obj/resource ∈ user-role permission) with reduced priviledge
=> Access = 1 iff (obj ∈ r_i . p_i)
Else level 3 access
End if.

//* **Level 3 access: *Deny User *//***
If (RV > φ_2)
Access = 0 (access denied).
End if.

4.0 Implementation and Evaluation

- The design was implemented using C# programming language which is part of .NET technology.
- The tests were performed on a computer system with Pentium IV 2.6 GHz Intel Processor, 1 GB RAM, 160GB hard disk, and running Microsoft Window 7 operating system.
- The model was evaluated in an academic environment where every user is expected to have submitted their data such as the user identification number, name and password into the database before login would be allowed.

Evaluation was carried out and the results obtained were classified into two:

(a) Result obtained from the security analysis
(b) Result obtained from the compliance level of CDTAC model as against existing models.

a) Result obtained from the security analysis

The security analysis relates to the issues of system evaluation.

(i) **System Evaluation:** This is done by setting up a network of 20 users. Each user is allowed to run through the system with various contexts and requests. For each user, the system decision is recorded and compared with the user's expected response.

The table below shows the result of the system response for 20 users who made certain requests with their expected results.

Table 1: Evaluation Result of the Access Control Model

User	Request	Context Parameters				Risk Value	Object Permission.	User's Expected Response	System Response
		R	L	T	F				
User 01	Edit lect. Note	0	0	1	1	0.250	1	Allow	Allow
User 02	View student result	0	0	1	0	0.125	1	Allow	Allow
User 03	Edit lecture note	0	0	0	0	0.000	1	Allow	Allow
User 04	Use Head of unit Printer	2	1	1	2	0.750	1	Deny	Deny
User 05	View staff record	1	1	1	0	0.375	0	Allow	Deny
User 06	Edit staff record	2	1	0	1	0.500	0	Deny	Deny
User 07	Use head of unit scanner	0	1	1	1	0.375	1	Allow	Allow
User 08	View std. file	0	1	0	1	0.25	1	Allow	Allow
User 09	Edit lecture note	2	1	1	1	0.625	0	Deny	Deny
User 10	View staff record	1	1	0	0	0.250	1	Allow	Allow
User 11	Edit lecture note	2	1	1	0	0.500	1	Allow	Allow
User 12	Edit student result	0	1	0	2	0.375	1	Allow	Allow
User 13	View student result	2	0	1	2	0.625	1	Deny	Deny
User 14	Use Head of unit Printer	0	0	0	1	0.125	0	Allow	Deny
User 15	Use head of unit scanner	0	0	1	1	0.250	0	Allow	Deny
User 16	View lecture note	2	0	0	0	0.250	1	Allow	Allow
User 17	View staff record	1	1	1	2	0.625	1	Deny	Deny
User 18	Edit student record	2	0	0	2	0.500	1	Allow	Allow
User 19	Edit student record	2	0	0	1	0.375	0	Allow	Deny
User 20	Edit student result	0	1	1	1	0.375	1	Allow	Allow

Keys: Object Permission [1, 0].

[1] implies that object/resource requested is contained in user's object permission.

[0] implies that object/resource requested is not part of the user's object permission. (R - role, L – location, T – time of access, and F – frequency of access).

The result from table 1 shows that five (5) out of the 21,200 attempts made were compromised-access and the possibility (P) of getting a compromised access is 0.00024. The associated compromised threat's value was observed to be insignificant which implies that the system developed does not give room for attack (i.e. unauthorized access) even when subjected to a threatened condition.

That is, P (of getting the compromised access) = $\frac{\sum \text{compromised attempts}}{\text{total number of attempts}}$

Therefore P(CA) = 5 / 21,200
= 0.00024 (not sig.)

To test for the Security of the System

The system was evaluated on twenty (20) randomly selected attempts from each node.

Security Index = No. of evaluated attempts – P(CA)
= 20 – 0.00024
= 19.999

Increase in average Security Index

= SI of the system – SI of existing system
= 19.999 – 19.81
= 0.18

Table 1 also shows that the CDTAC system is able to dynamically adjust to different requests based on their context parameters and can effectively protect network resources from abuse of privileges. Access decision is subject to user's permission which implies that irrespective of the level of the risk value, the requested object/resource must be contained in the permissions given to the user by the organization.

(b) Result obtained from the compliance level of CDTAC model as against existing models

The compliance level of CDTAC model as against existing models is the most important thing that affects the overall performance of the access control system. It addresses the issue of computational criteria required by the system to decide its decision which at the same time determine the security strength of the system. The CDTAC system is compared with other existing access control systems that are relevant with the model using the following criteria: (i) Environmental Context (ii) Authentication Protocol (iii) Inclusion of Information Security Objectives (iv) Defination of Standard policy (v) Risk Determination (vi) Threat Concentration (vii) Feedback and (viii) Cost are shown in table 2. The models in consideration are:

(a) CRAC ------ Contextual Risk-Based Access Control System (Nguyen et al., 2007)

(b) CRAAC ----- Context-Risk-Aware Access Control Model (Ali and Ning, 2008)

(c) CACA ----- Component-Based Access Control Architecture (Sodiya and Onasoga, 2009)

(d) GAPs ----- Towards Shrink-Wrapped Security: Practically Incorporating Context into Security Services (Gleenesha, 2011)

(e) CACCIS ---- Context-Aware Access Control for Clinical Information System (Ferhim, 2012).

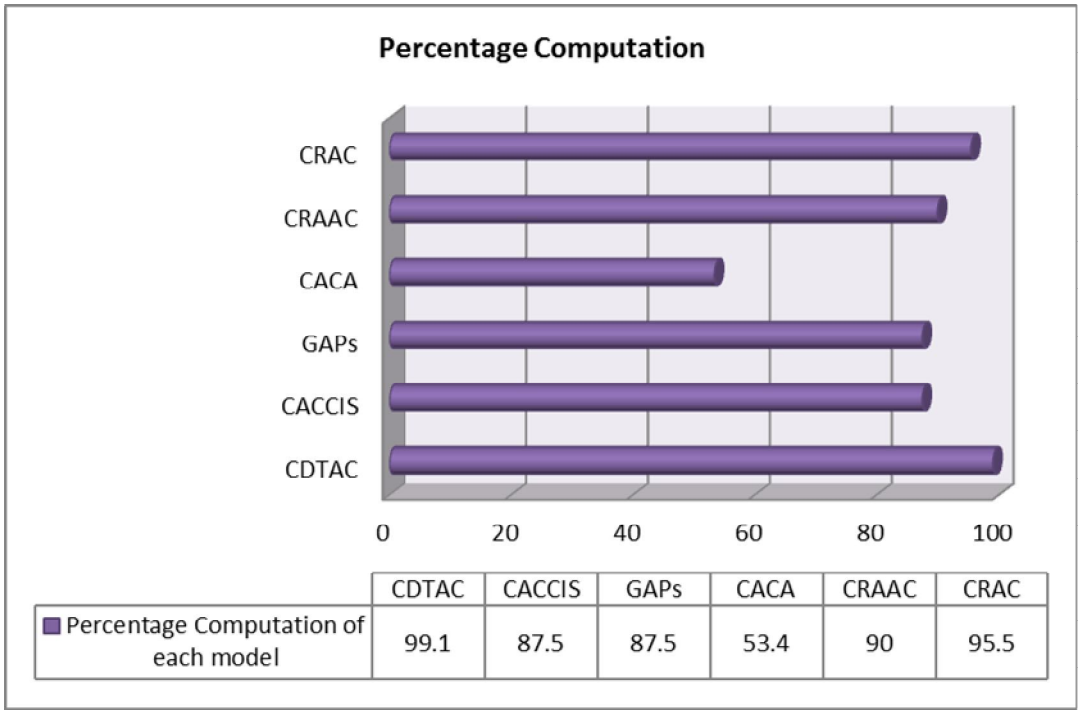
The comparison verifies the inclusion of any of the stated criteria in each of the aforementioned model.

Table 2: Comparison of Computational Overhead

CRITERIA	MODELS					
	CRAC	**CRAAC**	**CACA**	**GAPs**	**CACCIS**	**CDTAC**
Environmental Context	1	1	0	1	1	1
Authentication protocol	0	1	0	0	0	1
Information Security Objectives	1	0	0	1	0	1
Standard Policy Defination	0	1	1	0	1	1
Risk Determination	1	0	0	0	0	1
Environmental threat Concentration	0	0	0	0	0	1
Feedback	1	1	1	1	1	1
Cost	1	0	0	0	0	0

For every mentioned criteria, a numerical value one (1) is assigned for the availability of such criteria within that system while value zero (0) is assigned for its non-availability in the given model.

Figure 3 shows the trend of the percentage computation of the compliance level of each mentioned model based on the stated criteria.



	CDTAC	CACCIS	GAPs	CACA	CRAAC	CRAC
Percentage Computation of each model	99.1	87.5	87.5	53.4	90	95.5

Figure 3: Performance Evaluation Graph of CDTAC Against Other Models in Category

It is clear from figure 3 above that the CDTAC system has a maximum percentage of 99.10% of the policy compliance as compare to other models. The implication of this is that CDTAC system can protect network resources against unauthorized access and misuse of priviledges.

5.0 Conclusion and Future Works

In this paper, a new access control model, a Context Dependent Threat-Based Access Control (CDTAC) system is presented to solve security problem in access control mechanism. The model was able to quantify the context parameters that were not subject to bias, determine the threat level of the parameters in authentication phase, dynamically enforces the access policy on the user context parameters and automatically update the detailed information of current number of transaction/access by each user thereby reflecting the past behavior of the users with particular objects and also estimate the associated risk attached to every request before arriving at a valid access decision.

In future work, more parameters and factors that affect risk assessment process will be considered and the work shall be extended to intrusion detection and response system.

References

Al-Rwais S. & Al-Muhtadi J. (2010). A Context-aware Access Control Model for Pervasive Environment. Centre of Excellence in Information Assurance and the College of Computer and Information Sciences, King Saud University, Riyadh, Saudi Arabia. Year : 2010 | Volume : 27 | Issue : 5 | Page : 371-379.

Cheng P. C., Rohatgi P., Keser C., Karger P. A., Wagner G. M, & Reninger A. S. (2007). Fuzzy multi-level security: An experiment on quantified risk-adaptive access control. Technical Report RC24190, IBM Research, 2007.

CISA "Certified Information Systems Auditor". Net-Security.org. June 1, 2006. http://www.wikipedia.org/wiki/Certified_Information_Systems_Auditor.

Feinman T., Goldman, David, Wong, Ricky, Cooper & Neil (1999). Network security Resource. Pricewaterhouse Coopers LLP, Resource Protection Services, Security Basics: A White Paper, June 1, 1999.

Gleneesha J. (2010). Context-Aware Security", gjohnson@cs.umd.edu

Gleneesha J. et al. (2011). Towards Shrink Wrapped Security-Practically Incorporating Context Into Security Services. Procedia Computer Science 5, 782–787.

Guangsen Zhang (2004). Contextual Risk-based Access Control. http://www.uclab.khu.ac.kr/resources/publication/C_101.pdf.

Strembeck M., Neumann G. (2004). An integrated approach to engineer and enforce context constraints in RBAC. ACM Transactions on Information and System Security, 7 (3): 392-427

National Institute of Standards and Technology (NIST, 2006). Assessment of Access Control Systems. Interagency Report 7316. September, 2006.

NISTIR, National Institute of Standards and Technology Interagency Report, Assessment of Access Control Systems Reports on Computer Systems Technology. csrc.nist.gov/publications/nistir/7316/NISTIR-7316.pdf

Nguyen N. D., Sungyoung L., Young-Koo L. & Heejo L. (2007). Contextual Risk-Based Access Control. Security and Management 2007: pp 406-412.

Donn Parker B. (2011). Toward a New Framework for Information Security www.mekabay.com/courses/academic/.../csh5_ch03_parkerian_hexad.pptx. Copyright © 2011 M. E. Kabay.

Sodiya A. S. & Onashoga A. (2009). Components-Based Access Control Architecture. Department of Computer Science, University of Agriculture, Abeokuta, Nigeria; Issues in Informing Science and Information Technology, Volume 6, 2009.Sven O. H. (1994). Decision Theory: A Brief Introduction, Department of Philosophy and the History of Technology, Royal Institute of Technology (KTH), Stockholm.

Young-Kim C., Mon D., Jeong J., Lee C., Song D. & Baik (2005). Context aware access Control mechanism for ubiquitous applications. Advances in Web Intelligence, Springer Berlin/Heidelberg, May 2005, vol. 3528, pp. 236-242