## Studying Security Criteria's of Cloud and VM Platforms and Present New Innovative Solution for it

MEHDI DARBANDI

Department of Electrical Engineering and Computer Science at Iran University of Science and Technology, IRAN

Abstract: This paper categorized as such: in the first section of this paper, authors present general information about basis of cloud computing. By the means of several examples they show the importance of security in such networks. They discuss about security problems in VMware and combine it with security of cloud computing. After that they present their new novel approach for intensifying security of such network, they purpose Sequence Unscented Kalman Filtering Algorithm for acquiring better security. At the end of this paper, authors proof workability and results of applying such algorithm to the cloud computing by the means of mathematical demonstration and MATLAB simulations.

Keywords: Cloud Computing, Internet technology, Kalman filter.

Received: July 5, 2021. Revised: December 10, 2021. Accepted: December 22, 2021. Published: January 12, 2022.

#### **1. Introduction**

Infrastructure clouds make significant use of virtualization and the clouds provide computational resources which are consumed by the means of virtual machines [10]. Due to this strong connection between these two technologies, security problems associated with virtual machines will have an impact on the overall security of infrastructure clouds. Therefore a review of existing literature on the topic of virtual machine security will give us a useful foundation for analyzing the security of infrastructure clouds and provides an insight in the underlying security challenges [10]. Virtual machines provide a high degree of flexibility by allowing users to easily create, copy, snapshot, rollback, and migrate them. This flexibility results in major adoption of virtual machines by users for different purposes, e.g., for testing of

software or configurations using snapshots and the rollback mechanism [10]. The authors of the paper extracted the following list of security issues related to virtual machines. Scaling represents the problem that users now have multiple virtual machines, e.g., for testing and development purposes, instead of a very few number of physical ones [10]. Therefore the total number of machines drastically increases within one organization and the workload on the security systems will increases accordingly. Diversity in operating systems, OS versions and patch levels increases the complexity in the security management of the infrastructure [10]. VMs typically result in a high diversity, because users have multiple snapshots of VMs and testers can have a collection of different VMs. Transience is another security issue induced by the flexibility of virtual machines [10]. It

mainly deals with the problem that VMs appear and disappear very rapidly in the network which makes security management, e.g., patch management and vulnerability scanning, very difficult [10]. The authors describe this as the missing of a steady state in the network, where the steady state means that all machines are patched and properly managed. The Mobility of a VM, i.e., the VM can easily be copied or migrated, imposes multiple security problems: all the hosts, the VM will be executed on, have to be part of the trusted computing base (TCB) [10]; sensitive information can leave a security perimeter or malware is introduced, and the theft of VMs can easily be done by simply copying a file. The traditional Software Lifecycle, i.e., a monotonic forward progress of the software state, is broken by virtual machine's snapshot and rollback mechanisms, because the execution of the virtual machine can be forked and be rolled back [10]. In particular the rollback mechanism induces a lot of problems regarding freshness of randomness sources used for cryptographic protocols or critical patches are removed by a rollback. Limited Data Lifetime, e.g., for sensitive or cryptographic information, can be compromised due to the rollback mechanism and that the content of the virtual machine's memory might be stored on the disk of the host due to paging, snapshots, or migration [10]. In traditional computing environments, the Identity of a machine is often deduced from properties like the MAC address, the location, or Ethernet port. Virtual machines however use dynamically created MAC typically addresses and they might migrate from one physical host to another, therefore properties like the location or Ethernet port will change, and make it difficult to assign an identity [10]. Now, author of this paper present some

solutions and highlights some benefits about

Mehdi Darbandi

these technology [10]. The role of the VMM is to isolate the VMs from each other and the correctness of enforcing this property is crucial, therefore a high assurance VMM is required [10]. Introducing an extended virtualization layer that overtakes functionality originally performed in the guest operating systems has a certain number of benefits. Users do not have to worry about security management, e.g., firewalling or anti-virus detection, if these mechanisms are provided by the virtualization and are operated by a laver central administration staff. Furthermore. these security services are now independent of the guest operating systems, which results in a higher flexibility because a high diversity of VMs can be securely managed. Regarding the security issue associated with software lifecycle and the rollback feature, the virtualization layer could pro- vide mechanisms to store such sensitive information and to provide strong randomness [10].

The security of the Virtual Machine Monitor (VMM) is crucial, because it provides the necessary isolation between the hosted VMs and typically runs with the highest privileges on the system [10].

Introducing a new software layer, such as the one providing virtualization, inherently increases the complexity of the system, which also increases the possibility of software security vulnerabilities [10]. Such vulnerabilities in the VMM can lead to the break of isolation, i.e., a VM can access other VMs resources. Different solutions exist to mitigate security problems in the virtualization layer which are based on principles of building secure software [10]: formal verification, security by isolation and disaggregation, and reducing the trusted code size [10].

An interesting example for formal verification of software, which is also relevant for our topic of VMM security, is the seL4 project, a formally verified L4 microkernel. The proof verifies that the implementation in the C programming languages matches the abstract specification of the system and implies that certain software vulnerabilities, like buffer overflows and null pointer dereferences, are absent in the implementation. Microkernel and VMM are very similar, therefore either a formally verified microkernel acting as a VMM can be used or adapting the formal proof for VMMs, although the size of existing VMMs make formal verification very difficult. Another approach of improving the security of the VMM is to reduce the complexity and trusted code base (TCB) by means of decomposition. An approach for extracting the domain builder functionality of the Xen dom0 into a separate domain. With a separate domain builder VM, the user-space of dom0 can be removed from the TCB, because no privileged functionality for VM construction and management need to be exposed to user-space applications, e.g., xend. However, in their current state the dom0 kernel is still part of the TCB due to required interaction with physical I/O devices. Besides the dom0 kernel, the Xen hypervisor and the domain builder are part of the TCB [10].

The recent prototype operating system Qubes OS implements, among other security features, disaggregation of Xen dom0 by establishing driver domains which are limited to a specific hardware resource by the means of IOMMU as implemented by Intel VT-d, i.e., the VMM monitors DMA requests and can possibly restrict them [10]. These driver domains can run with limited privileges and the overall complexity of dom0 can be reduced. Thereby software vulnerability in one of the drivers will not result in a break of isolation when running in a non-privilege driver domain compared to running in dom<sup>0</sup>. IOMMU would also benefit the disaggregation of Xen using a domain builder VM, because the dom0 kernel could be removed from the TCB when I/O with physical devices is offloaded to driver domains. The virtualization architecture NOVA uses a minimal microkernel, with a size similar to the formallv verified seL4. and provides virtualization functionality user-land as applications. Therefore the amount of high privilege code is reduced to a minimal microkernel-based hypervisor [10].

## 2. Sequence Unscented Kalman Filtering Algorithm:

Since Unscented Kalman Filter (UKF) is proposed by Julier and Uhlman, it has absolved many researchers to study it, and many kinds of new algorithms [2-6] with different accuracies have come out. Unlike the Extend Kalman Filter (EKF), which is based on the linearizing the nonlinear function by using Taylor series expansions, UKF uses the true nonlinear models and approximates a distribution of the state random variable [11]. Furthermore, it only needs a minimal set carefully chosen sample points, by which the posterior mean and covariance can be accurate to the second order for any nonlinearity, avoiding Jacobian's computation. If the priori random variable is Gaussian, the estimation of the posterior mean and covariance can could be accurate to the third order. It can be seen that in all the UKF algorithms it needs inversing the matrix in measurement update. The dimension of the inversed matrix is equal to that of measurement vector. If the dimension of the measurement is very large, so it could cost a great computing time. In order to decrease the computing cost and not to inverse the matrix, a sequence method is used to solve this problem [11].

In this paper, the sequence UKF is proposed. It deals with nonlinear stochastic system with

linear measurement. Based on RBUKF and traditional Kalman Filter (KF), it deduced the special algorithm for the sequence UKF in case of the covariance matrix of measurement noise is diagonal matrix or not. In theory it is proven that the sequence UKF has the same estimation accuracy with RBUKF, but has lower computational cost. Simulation results verify the high performance of sequence UKF.

## 3. UKF Mechanism:

UKF is used to solve the estimation problem for any nonlinear system. The considered nonlinear system is represented by [11]:

$$\begin{cases} \mathbf{x}_{k+1} = \mathbf{f}(\mathbf{x}_k) + \mathbf{v}_k \\ \mathbf{z}_k = \mathbf{h}(\mathbf{x}_k) + \mathbf{w}_k \end{cases}$$

Where  $\mathbf{x}_k$  and  $\mathbf{z}_k$  denote the state vector with ndimension and the measurement vector with m-dimension at step k, respectively. The nonlinear mapping  $f(\cdot)$  and  $h(\cdot)$  are assumed to be continuously differentiable with respect to  $\mathbf{x}_k$ . Moreover,  $\mathbf{v}_k \sim N(\mathbf{v}_k; \mathbf{0}, \mathbf{Q}_k)$  denote the *n*-dimension. with process noise  $\mathbf{w}_k \sim N(\mathbf{w}_k; \mathbf{0}, \mathbf{R}_k)$  denote the measurement with m -dimension.  $\mathbf{v}_k$  and  $\mathbf{W}_k$ noise are independent of each other.

Like Kalman Filter (KF), UKF is also a minimum mean-square error estimator (MMSE). For system (1), the mechanism of MMSE is time-update and measurement-update as follows [11]:

Time-update:

$$\mathbf{x}_{k/k-1} = E[\mathbf{f}(\mathbf{x}_{k-1})]$$
$$\mathbf{P}_{k/k-1} = E[\mathbf{e}_k \mathbf{e}_k^T]$$
Measurement-update:
$$\hat{\mathbf{x}}_k = \mathbf{x}_{k/k-1} + \mathbf{W}_k \mathbf{v}_n$$
$$\mathbf{v}_k = \mathbf{z}_k - \hat{\mathbf{z}}_k$$
$$\hat{\mathbf{z}}_k = E[\mathbf{h}(\mathbf{x}_k)]$$

 $\mathbf{P}_k = \mathbf{P}_{k/k-1} - \mathbf{W}_k \mathbf{S}_k \mathbf{W}_k^T$ 

Where  $\mathbf{e}_k = \mathbf{x}_k - \mathbf{x}_{k/k-1}$ , the weight matrix  $\mathbf{W}_k$  is chosen to minimize the trace of the updated covariance  $\mathbf{P}_k$ . Its value is calculated from [11]:  $\mathbf{W}_k = \mathbf{P}_k^{xz} \mathbf{S}_k^{-1}$ 

Where  $\mathbf{P}_{k}^{xz}$  is the cross covariance between  $\mathbf{e}_k$  and  $\mathbf{v}_k$ ,  $\mathbf{S}_k$  is the covariance of  $\mathbf{v}_k$ . UKF is based on the mechanism above. By applying the unscented transformation (UT) to number of chosen sigma а points,  $\mathbf{x}_{k/k-1}$ ,  $\mathbf{P}_{k/k-1}$ ,  $\mathbf{S}_k$  and  $\mathbf{P}_k^{xz}$  can be approximately expressed by the linear composition of the transformed sigma points. So UKF solves the nonlinear estimation problem using MMSE mechanism. When measurement equation in system is linear, it changes to system as follows [11]:

$$\begin{cases} \mathbf{x}_{k+1} = \mathbf{f}(\mathbf{x}_k) + \mathbf{v}_k \\ \mathbf{z}_k = \mathbf{H}_k x_k + \mathbf{w}_k \end{cases}$$

To solve the estimation problem for system, it only needs transformed sigma points to estimate  $\mathbf{x}_{k/k-1}$  and  $\mathbf{P}_{k/k-1}$ , and  $\mathbf{S}_k$  and  $\mathbf{P}_k^{xz}$  can be computed accurately. So the UKF algorithm can be reduced to RBUKF algorithm. Compared to UKF, RBUKF is not less computational cost, but also higher accuracies. The difference part between the UKF and RBUKF is measurement-update. For comparison, it only gives the measurementupdate of RBUKF as follows [11]:

$$\mathbf{P}_{zz,k} = \mathbf{H}_{k} \mathbf{P}_{k|k-1} \mathbf{H}_{k}^{T} + \mathbf{R}_{k}$$
$$\mathbf{P}_{k,xz} = \mathbf{P}_{k|k-1} \mathbf{H}_{k}^{T}$$
$$\mathbf{K}_{k} = \mathbf{P}_{xz,k|k-1} \mathbf{P}_{zz,k}^{-1}$$
$$\hat{\mathbf{x}}_{k} = \mathbf{x}_{k|k-1} + \mathbf{K}_{k} (\mathbf{z}_{k} - \mathbf{z}_{k|k-1})$$
$$\mathbf{P}_{k} = \mathbf{P}_{k|k-1} - \mathbf{K}_{k} \mathbf{P}_{zz,k} \mathbf{K}_{k}^{T}$$

## 4. Sequence UKF Algorithm

No matter UKF or RBUKF, it needs to inverse the matrix in measurement-update. If the dimension of the measurement vector is very large, it could be a great computational cost for compute gain matrix  $\mathbf{K}_k$ . In order to avoid inversing the matrix in computing  $\mathbf{K}_k$ , the sequence UKF will deal with ever component of measurement vector one by one instead of the vector at one time. This method needs not to inverse the matrix and can greatly decrease computational cost. For system, it deduced the sequence UKF as follows [11].

**Theory I**: For system, the measurementupdate in UKF can be computed as follows:

$$\mathbf{K}_{k}^{i} = \mathbf{P}_{k}^{i-1}\mathbf{H}_{k}^{iT} (\mathbf{H}_{k}^{i}\mathbf{P}_{k}^{i-1}\mathbf{H}_{k}^{iT} + R_{k}^{i})^{-1}$$
$$\mathbf{x}_{k}^{i} = \mathbf{x}_{k}^{i-1} + \mathbf{K}_{k}^{i} (z_{k}^{i} - \mathbf{H}_{k}^{i}\mathbf{x}_{k}^{i-1})$$
$$\mathbf{P}_{k}^{i} = \mathbf{P}_{k}^{i-1} - \mathbf{K}_{k}^{i}\mathbf{H}_{k}^{i}\mathbf{P}_{k}^{i-1} \quad i = 1, 2, \cdots, n$$

Where  $\mathbf{H}_{k}^{i}$  is the *i*-th row in  $\mathbf{H}_{k}$ ,  $z_{k}^{i}$  is the *i*-th scalar in the measurement vector at step *k*,  $R_{k}^{i}$  is the *i*-th element in diagonal of  $\mathbf{R}_{k}$  [11].

#### **Proof:**

Rewrite the measurement equation in system, it gets [11]:

$$\begin{bmatrix} z_k^1 \\ z_k^1 \\ \vdots \\ z_k^m \end{bmatrix} = \begin{bmatrix} \mathbf{H}_k^1 \\ \mathbf{H}_k^2 \\ \vdots \\ \mathbf{H}_k^m \end{bmatrix} \mathbf{x}_k + \begin{bmatrix} v_k^1 \\ v_k^2 \\ \vdots \\ v_k^m \end{bmatrix}$$

Because of the form of, the measurement vector  $\mathbf{z}_k$  can be seen as  $z_k^i$  ( $i = 1, 2, \dots, m$ ) one by one to update the time-update equations. It must be noticed that when  $z_k^i$  one by one instead of  $\mathbf{z}_k$  updates the equations in measurement-update, it implies that the state equation is invariant for every  $z_k^i$  at step k. So the measurement-update in UKF at step k is equivalent to the filtering problem of new linear systems as follows [11]:

$$\begin{cases} \mathbf{x}_k^i = \mathbf{x}_k^{i-1} \\ z_k^i = \mathbf{H}_k^i \mathbf{x}_k^i + w_k^i \quad (i = 1, 2, \cdots, m) \end{cases}$$

Where  $w_k^i \sim N(w_k^i; 0, R_k^i)$ , is the equivalent measurement noise. The initial value of the filter is  $\mathbf{x}_k^0 = \mathbf{x}_{k/k-1}$  and  $\mathbf{P}_k^0 = \mathbf{P}_{k/k-1}$ .

So the equivalent linear filter can be easily derived from classic Kalman Filter equations as follows:

$$\mathbf{x}_{k}^{i/i-1} = \mathbf{x}_{k}^{i-1}$$

$$\mathbf{x}_{k}^{i} = \mathbf{x}_{k}^{i/i-1} + \mathbf{K}_{k}^{i} (z_{k}^{i} - \mathbf{H}_{k}^{i} \mathbf{x}_{k}^{i/i-1})$$

$$\mathbf{K}_{k}^{i} = \mathbf{P}_{k}^{i/i-1} \mathbf{H}_{k}^{iT} (\mathbf{H}_{k}^{i} \mathbf{P}_{k}^{i/i-1} \mathbf{H}_{k}^{iT} + \mathbf{R}_{k}^{i})^{-1}$$

$$\mathbf{P}_{k}^{i/i-1} = \mathbf{IP}_{k}^{i-1} \mathbf{I}^{T}$$

$$\mathbf{P}_{k}^{i} = (\mathbf{I} - \mathbf{K}_{k}^{i} \mathbf{H}_{k}^{i}) \mathbf{P}_{k}^{i/i-1}$$

$$\mathbf{Or} (\mathbf{P}_{k}^{i})^{-1} = (\mathbf{P}_{k}^{i/i-1})^{-1} + \mathbf{H}_{k}^{iT} (\mathbf{R}_{k}^{i})^{-1} \mathbf{H}_{k}^{i}$$

Substituting these recent equations results in the following, and substituting previous equations in each other, results in next equations. So it obtains the new equations for the measurement-update in UKF as follows:

$$\mathbf{x}_{k}^{i} = \mathbf{x}_{k}^{i-1} + \mathbf{K}_{k}^{i} (z_{k}^{i} - \mathbf{H}_{k}^{i} \mathbf{x}_{k}^{i-1})$$
$$\mathbf{K}_{k}^{i} = \mathbf{P}_{k}^{i-1} \mathbf{H}_{k}^{iT} (\mathbf{H}_{k}^{i} \mathbf{P}_{k}^{i-1} \mathbf{H}_{k}^{iT} + \mathbf{R}_{k}^{i})^{-1}$$
$$\mathbf{P}_{k}^{i} = (\mathbf{I} - \mathbf{K}_{k}^{i} \mathbf{H}_{k}^{i}) \mathbf{P}_{k}^{i-1}$$
$$\mathbf{Or} \quad (\mathbf{P}_{k}^{i})^{-1} = (\mathbf{P}_{k}^{i-1})^{-1} + \mathbf{H}_{k}^{iT} (\mathbf{R}_{k}^{i})^{-1} \mathbf{H}_{k}^{i}$$

By theory I and the UKF mechanism, it obtains the sequence UKF algorithm as follows: Calculate sigma points [11]:

$$\begin{cases} \chi_{i,k-1} = \mathbf{x}_{i,k-1} & i = 0, \\ \chi_{i,k-1} = \mathbf{x}_{i,k-1} + (\sqrt{(L+\lambda)}\mathbf{P}_{k-1})_i & i = 1, \cdots, L, \\ \chi_{i,k-1} = \mathbf{x}_{i,k-1} - (\sqrt{(L+\lambda)}\mathbf{P}_{k-1})_{i-L} & i = L+1, \cdots, 2L, \\ \end{cases}$$

e-update:

$$\begin{aligned} \boldsymbol{\chi}_{k/k-1} &= \mathbf{f}(\boldsymbol{\chi}_{k-1}) \\ \mathbf{x}_{k/k-1} &= \sum_{i=0}^{2n} W_i^{(m)} \boldsymbol{\chi}_{i,k/k-1} \\ \mathbf{P}_{k/k-1} &= \sum_{i=0}^{2n} W_i^{(c)} (\boldsymbol{\chi}_{i,k/k-1} - \mathbf{x}_{k|k-1}) (\boldsymbol{\chi}_{i,k/k-1} - \mathbf{x}_{k/k-1})^T + \mathbf{Q}_k \\ z_{k/k-1}^i &= \mathbf{H}_k^i \mathbf{x}_{k/k-1}^i \quad (i = 1, 2, \cdots, m) \end{aligned}$$

Measurement-update:

$$\mathbf{P}_{k}^{0} = \mathbf{P}_{k/k-1}$$

$$\mathbf{x}_{k}^{0} = \mathbf{x}_{k/k-1}$$

$$\mathbf{x}_{k}^{i} = \mathbf{x}_{k}^{i-1} + \mathbf{K}_{k}^{i} (z_{k}^{i} - \mathbf{H}_{k}^{i} \mathbf{x}_{k}^{i-1})$$

$$\mathbf{K}_{k}^{i} = \mathbf{P}_{k}^{i-1} \mathbf{H}_{k}^{iT} (\mathbf{H}_{k}^{i} \mathbf{P}_{k}^{i-1} \mathbf{H}_{k}^{iT} + R_{k}^{i})^{-1}$$

$$\mathbf{P}_{k}^{i} = (\mathbf{I} - \mathbf{K}_{k}^{i} \mathbf{H}_{k}^{i}) \mathbf{P}_{k}^{i-1} (i = 1, 2, \cdots, m)$$

$$\mathbf{x}_{k} = \mathbf{x}_{k}^{m}$$

$$\mathbf{P}_{k} = \mathbf{P}_{k}^{m}$$
Where  $W_{0}^{(c)} = \lambda/(L + \lambda) + (1 - \alpha^{2} + \beta)$ ,  
 $W_{i}^{(m)} = W_{i}^{(c)} = 1/[2(L + \lambda)] \quad i = 1, 2, \cdots, 2L$ 

# **5. Algorithm Performance Analysis:**

#### Filtering accuracy Analysis:

For system, in order to comparison of RBUKF and the sequence UKF, it needs to get the filtering covariance respectively. Firstly, it is to

derived covariance  $\mathbf{P}_k$  of RBUKF. By definition, it calculates covariance  $\mathbf{P}_k$  of

RBUKF as follows:

 $\mathbf{P}_{k} = E[(\mathbf{x}_{k} - \hat{\mathbf{x}}_{k})(\mathbf{x}_{k} - \hat{\mathbf{x}}_{k})^{T}]$ 

Substitute this equation in previous equations results in [11]:

$$\mathbf{P}_{k} = \mathbf{P}_{k/k-1} - \mathbf{P}_{k/k-1} \mathbf{H}_{k}^{T} \mathbf{K}_{k}^{T} - \mathbf{K}_{k} \mathbf{H}_{k} \mathbf{P}_{k/k-1}$$
$$+ \mathbf{K}_{k} \mathbf{H}_{k} \mathbf{P}_{k/k-1} \mathbf{H}_{k}^{T} \mathbf{K}_{k}^{T}$$

By previous equations, it obtains

$$\mathbf{K}_{k} = \mathbf{P}_{xz,k/k-1} \mathbf{P}_{zz,k/k-1}^{-1}$$
$$= \mathbf{P}_{k/k-1} (\mathbf{H}_{k} \mathbf{P}_{k/k-1} \mathbf{H}_{k}^{T} + \mathbf{R}_{k})^{-1}$$

Substituting these recent equations results in:

$$\mathbf{P}_{k} = \mathbf{P}_{k/k-1} - \mathbf{P}_{k/k-1} \mathbf{H}_{k}^{T} (\mathbf{H}_{k} \mathbf{P}_{k/k-1} \mathbf{H}_{k}^{T} + \mathbf{R}_{k})^{-1} \mathbf{H}_{k} \mathbf{P}_{k/k-1}$$
$$= (\mathbf{P}_{k/k-1}^{-1} + \mathbf{H}_{k}^{T} \mathbf{R}_{k}^{-1} \mathbf{H}_{k})^{-1}$$

By inversing both sides of this equation, it obtains [11]:

$$\mathbf{P}_{k}^{-1} = \mathbf{P}_{k-1/k}^{-1} + \mathbf{H}_{k}^{T} \mathbf{R}_{k}^{-1} \mathbf{H}_{k}$$

Because of:

$$\mathbf{H}_{k} = [\mathbf{H}_{k}^{T} \ \mathbf{H}_{k}^{2T} \cdots \mathbf{H}_{k}^{mT}]^{T}$$
$$\mathbf{R}_{k}^{-1} = (diag(R_{k}^{1} \ R_{k}^{2} \cdots R_{k}^{m}))^{-1}$$
$$= diag((R_{k}^{1})^{-1} \ (R_{k}^{2})^{-1} \cdots (R_{k}^{m})^{-1})$$

Substituting these recent equations in each other, and doing matrices multiplication, it gets  $\mathbf{P}_{k}^{-1} = \mathbf{P}_{k-1/k}^{-1} + \mathbf{H}_{k}^{T} \mathbf{R}_{k}^{-1} \mathbf{H}_{k}$ 

$$= \mathbf{P}_{k-1/k}^{-1} + \sum_{i=1}^{m} \mathbf{H}_{k}^{iT} (\mathbf{R}_{k}^{i})^{-1} \mathbf{H}_{k}^{i}$$

Secondly, it calculates the covariance of the sequence UKF. By the measurement-update equation, it easily gets

$$(\mathbf{P}_{k}^{m})^{-1} = (\mathbf{P}_{k}^{m-1})^{-1} + \mathbf{H}_{k}^{mT} (\mathbf{R}_{k}^{m})^{-1} \mathbf{H}_{k}^{m}$$

$$= (\mathbf{P}_{k}^{m-2})^{-1} + (\mathbf{H}_{k}^{m-1})^{T} (\mathbf{R}_{k}^{m-1})^{-1} \mathbf{H}_{k}^{m-1}$$

$$+ \mathbf{H}_{k}^{mT} (\mathbf{R}_{k}^{m})^{-1} \mathbf{H}_{k}^{m}$$

$$\cdots$$

$$= (\mathbf{P}_{k}^{0})^{-1} + \sum_{i=1}^{m} (\mathbf{H}_{k}^{i-1})^{T} (\mathbf{R}_{k}^{i-1})^{-1} \mathbf{H}_{k}^{i-1}$$

Substituting previous equations in this equation results in the covariance of the sequence UKF as follows:

$$(\hat{\mathbf{P}}_{k})^{-1} = = (\mathbf{P}_{k/k-1})^{-1} + \sum_{i=1}^{m} (\mathbf{H}_{k}^{i})^{T} (\mathbf{R}_{k}^{i})^{-1} \mathbf{H}_{k}^{i}$$

Compare these two recent equations, it can be seen that the covariance of the sequence UKF is equal to that of RUKF, which means that the accuracies of the two filters are the same in theory [11].

#### Computational Complexity Analysis:

In the sequence UKF algorithm, because  $\mathbf{H}_{k}^{i}\mathbf{P}_{k}^{i-1}\mathbf{H}_{k}^{iT} + R_{k}^{i}$  is a scalar, so this algorithm has successfully converted the inversion of mdimension matrix into m times division. The computational complexity has been greatly order to decreased. In compare the computational complexity between the sequence UKF and RBUKF, for their timeupdate algorithms are the same, here only gives the comparison results of their measurementupdate algorithms in table I. From this table, it can be seen that number of calculating times in RBUKF algorithm contains the third order of measurement dimension, while the sequence UKF has only second order component. So when the measurement dimension is large, the computational cost in the sequence UKF will be much less [11].

TABLE I

CALCULATION TIMES COMPARISO	N
-----------------------------	---

Algorith	Num of $\times \div$	Num of+-
т		
The	$(5m-1)n^2 + 2m^2n$	$5mn^2 + (m^2 - 4m)n$
sequence	$+\frac{2}{3}m^{3}+m^{2}+\frac{4}{3}m$	$+m^{3}/_{2}+m/_{2}$
UKF		/2/2
RBUKF	$4mn^2 + 4mn + m$	$4mn^2$

#### 6. Numerical Simulation:

In order to show the efficiency of the sequence UKF, it is applied to an example system in comparison with the RBUKF. Estimation performance and computational complexity of the filters are evaluated with Monte Carlo simulations [11].

The numerical example considered in this section is a fifth-order nonlinear model given by system, with four-dimension measurement.

 ${\bf x}_{k+1} =$ 

$$\mathbf{H}_{k} = \begin{bmatrix} \sin x_{1,k} \cos x_{2,k} + 0.5x_{2,k} - 0.1(x_{3,k})^{2} \\ \sin x_{1,k} + (\sin x_{2,k})^{2} - 0.1x_{5,k} \\ \cos x_{1,k} + \exp(-x_{2,k}) + 0.1(x_{3,k})^{2} \\ \sin x_{5,k} + \cos^{2}(x_{4,k}) - 0.5x_{2,k} \\ \sin x_{4,k} + \cos^{2}(x_{3,k}) - 0.1x_{1,k} \end{bmatrix} + \begin{bmatrix} v_{1,k} \\ v_{2,k} \\ v_{3,k} \\ v_{4,k} \\ v_{5,k} \end{bmatrix}$$

The covariance matrices of  $\mathbf{v}_k$  and  $\mathbf{w}_k$  are:  $\mathbf{Q}_k = 0.0001 \mathbf{I}_5$ ,  $\mathbf{R}_k = 0.01 \mathbf{I}_4$ .

The initial conditions for the system and the filters are [11]:

$$x_{1,0} = x_{2,0} = x_{3,0} = x_{4,0} = x_{5,0} = 0.5$$
$$\hat{x}_{1,0} = \hat{x}_{2,0} = \hat{x}_{3,0} = \hat{x}_{4,0} = \hat{x}_{5,0} = 2$$

And the initial covariance matrix is chosen as



Fig.3 comparison of computational complexity

For briefness, it only shows the filtering results of state  $x_5$ . From fig. 1, it can be seen the filtering accuracy of the sequence UKF is same with that of RBUKF, but curve of the RBUKF is more fluctuant than that of the sequence UKF, which means that the sequence UKF maybe has a better filtering performance in actually usage. Fig. 2 is the filtering covariance of the two filters. It can be seen that the filtering covariance is almost equal to each other. Fig. 3 is the comparison of the computational complexity between the two filters for the example. It shows that the number of multiplication and division in the sequence UKF is much smaller than that in RBUKF, so is it for the number of addition and subtraction. It verifies that the computational complexity of sequence is less than RBUKF. For the computational cost of RBUKF is less than traditional UKF. So the sequence UKF will have great advantage in comparison with UKF [11].

## 7. Conclusion:

In this paper, at first authors discuss about different aspects of cloud computing. They discuss about literature and principles of this technology. After that, they highlight the security of such network and remark this important factor by the means of several examples. After that, authors discuss about security problems of VMware's and cloud infrastructures and by purposing new generation of Kalman filter they attempt to overcome to this problem. Authors claim that by using of such algorithm like Kalman filter (that is based on knowing past and present states and predict future state according to them) they can estimate and predict about lots of important factors and avoid lots of crashes.

#### References

- [1] Mehdi Darbandi "Applying Kalman Filtering in solving SSM estimation problem by the means of EM algorithm with considering a practical example"; published by the Journal of Computing – **Springer**, 2012; USA.
- [2] Mehdi Darbandi; "Comparison between miscellaneous platforms that present for cloud computing and accreting the security of these platforms by new filter"; published by the Journal of Computing – **Springer**, 2012; USA.

- [3] Mehdi Darbandi; "New and novel technique in designing electromagnetic filter for eliminating EMI radiations and optimization performances"; published by the Journal of Computing - **Springer**, 2012; USA.
- [4] Mehdi Darbandi; "Developing concept of electromagnetic filter design by considering new parameters and use of mathematical analysis"; published by the Journal of Computing - **Springer**, 2012; USA.
- [5] Mehdi Darbandi; "Is the cloud computing real or hype Affirmation momentous traits of this technology by proffering maiden scenarios"; published by the Journal of Computing – Springer, 2012; USA.
- [6] Mehdi Darbandi; "Measurement and collation overriding traits of computer networks and ascertainment consequential exclusivities of cloud computing by the means of Bucy filtering"; published by the Journal of Computing - **Springer**, 2012; USA.
- [7] Mehdi Darbandi; "Unabridged collation about multifarious computing methods and outreaching cloud computing based on innovative procedure"; published by the Journal of Computing - **Springer**, 2012; USA.
- [8] Mehdi Darbandi; "Scrutiny about all security standards in cloud computing and present new novel standard for security of such networks"; published by the Journal of Computing - Springer, 2012; USA.
- [9] Microsoft's Accessible Technology Vision and Strategy; September 2011.
- [10] MSc. Thesis of Sören Bleikertz; Norwegian University of Science and Technology; June 2010.
- [11] Hui-ping Li, De-min Xu and Fu-bin Zhang ; "Sequence Unscented Kalman Filtering Algorithm".
- [12] Mehdi Darbandi; "Appraising the role of cloud computing in daily life and presenting new solutions for stabilization of this technology"; published by the Journal of Computing - Springer, 2012; USA.

## **Creative Commons Attribution License 4.0** (Attribution 4.0 International, CC BY 4.0)

This article is published under the terms of the Creative Commons Attribution License 4.0 <u>https://creativecommons.org/licenses/by/4.0/deed.en\_US</u>