Using Context in Security Design of a Search and Rescue System

Shivakant Mishra

Department of Computer Science
University of Colorado, Boulder, CO 80309-0430, USA
mishras@cs.colorado.edu

Abstract. With the emergence of small devices equipped with wireless communication, several sophisticated systems for search and rescue have been proposed and developed. However, a key obstacle in a wide deployment of these systems has been users' security and privacy. On one hand, such systems need to collect as much information about a user as possible in order to locate that user in a timely manner. On the other hand, this very capability drives users away from using such a system. This paper describes work-in-progress in building a security and privacy framework for CenWits, which is a new search and rescue system for people in emergency situation in wilderness areas. The paper focuses on the role of context in building this framework.

1 Introduction

Search and rescue of people in emergency situation in a timely manner is an extremely important service. In the past, it was difficult to build such a service because of a lack of timely information needed to determine the current location of a person who may be in an emergency situation. However, with the emergence of small computing devices such as PDAs, sensors and cell phones that have wireless communication capabilities, it has become feasible to build such a system. Indeed, several such systems have been proposed and prototypes of some of them have been implemented over the last five years [12, 1, 5, 6, 4].

We have designed and implemented a search and rescue system called CenWits[21] for a wilderness environment. A key differentiating feature of CenWits from the other recent search and rescue systems is that it is designed for a wilderness environment. In such an environment, there is no Internet connectivity, no cellular network, and building an adhoc network is infeasible due to an extremely sparse environment. Furthermore, GPS reception is only available at limited areas. CenWits (**C**onnection-less S**en**sor-Based Tracking System Using **Wit**nesses) is comprised of three components: (1) mobile, in-situ sensors that are worn by people (e.g. hikers); (2) access points (AP) that collect information from these sensors; and (3) GPS receivers and location points (LP) that provide location information to the sensors. A subject uses GPS receivers (when it can connect to a satellite) and LPs to determine its current location. The key idea of CenWits is that it uses a concept of *witnesses* to convey a subject's movement and location information to the outside world. This averts a need for maintaining a connected network to transmit location information to the outside world. In particular, there is no need for expensive GSM or satellite transmitters, or maintaining an adhoc network of in-situ sensors in CenWits.

Since a search and rescue system like CenWits must track the movement of people, there are some very obvious and important security and privacy issues. In fact, such systems must cope with two conflicting issues. On one hand, the system requires a collection of as much information about the location and movement of a person as possible. This is to ensure that a smaller and more accurate search area may be determined in case that person goes missing, or is in emergency situation. Indeed, it is in the interest of a person to give out as much information as possible about his/her location and movement to improve his/her chances of being located and rescued in case of emergency. On the other hand, a majority of people are not comfortable in giving out too much information about their location and movement for the fear that such information may be misused for malicious purposes, e.g. stalking. Indeed, this latter reason has proved to be a major hindrance in a wider deployment of CenWits.

It is clear that appropriate security and privacy support must be provided in a search and rescue system for wide acceptance. At present, we are designing a security framework for CenWits. In this paper, we describe a preliminary design of this framework, and discuss some important issues in the design and implementation of security and privacy support for a search and rescue system in general. An important observation is that there is no single security model that can be applied in such a system. The required security and privacy support varies based on individuals as well as context. In this paper, we focus on the role of context in the design of a security framework.

The rest of this paper is organized as follows. Section 2 provides a brief overview of CenWits. Section 4 describes the role of context in building a security and privacy framework for CenWits. Section 5 provides a high-level description of a multi-level security and privacy framework for CenWits. Finally, Section 6 concludes the paper.

2 CenWits: A Brief Overview

CenWits is a search and rescue system that makes use of smaller and cheaper sensor devices. It has several important advantages over the other search and rescue systems. These advantages include a loosely-coupled system that relies only on intermittent network connectivity, power and storage efficiency, and low cost. It solves one of the greatest problems plaguing modern search and rescue systems: it has an inherent on-site storage capability. This means someone within the network will have access to the last-known-location information of a victim, and perhaps his bearing and speed information as well. It utilizes the concept of witnesses to propagate information, infer current possible location and speed of a subject, and identify hot search and rescue areas in case of emergencies.

The concept of witness works as follows. Whenever two or more hikers are with in a close range (say 100 meters) of one another, their sensors can exchange messages over a radio frequency. When two hikers, say A and B are in close range of each other, the following message exchange takes place. A generates a *witness record* that stores the following information: B was seen at this location at this time. Similarly, B generates a witness record storing A was seen at this location at this time. In addition, A sends all witness records in his/her memory to B, and similarly, B send all witness records in his/her memory to A. Whenever a hiker comes in close range of an access point (special computing devices that have Internet connection to a control center), he/she dumps all witness records in his/her memory to the access point.

A prototype of CenWits has been implemented using MICA2 sensor 900MHz running Mantis OS 0.9.1b. We have experimented with it in a number of indoor and outdoor environments.

3 Related Work

A survey of location systems for ubiquitous computing is provided in [11]. These include [17], [19], [8], and [1]. These systems are mainly designed for an indoor environment, and not useful for our purpose. A system that is viable in suburban area where a user can see clear sky and has GSM cellular reception at the same time is [5]. Since, cellular reception is not available in wilderness areas, this system is not useful for our purpose.

Personal Locater Beacon (PLB) that uses RF transmitter has been used for avalanche rescuing for years. Luxury version of PLB that combines a GPS receiver and a COSPAS-SARSAT satellite transmitter this also available [4]. Both of these devices are impractical in wilderness environment because of significantly large space and/or expensive satellite transmitter. Another related technology in widespread use today is the ONSTAR system [3], typically used in several luxury cars. Like PLBs, this system has several limitations for use in a wilderness environment, including heavy-weight, expensive, and requirement for a connected network. The Lifetch system uses a GPS receiver board combined with a GSM/GPRS transmitter and an RF transmitter in one wireless sensor node called Intelligent Communication Unit (ICU). Again, this system requires a connected network, which is not possible in a wilderness environment.

As far as we know, there is no work done yet in the area of building a security and privacy framework for a search and rescue system in a wilderness environment. The main difficulty in building this framework is the absence of any kind of infrastructure, be it a communication network or a public key infrastructure. Furthermore, the actual level of security and privacy needed in this environment varies based on circumstances as well as individuals' perception of danger. Our goal is make use of existing security principles and techniques, and adapt them meet the requirements of our system.

4 Motivation and Context Awareness

The amount of security and privacy support that a search and rescue system in a wilderness area should provide at any specific moment is strongly context dependent. There are two types of contexts involved here: situational and personal. Situational context refers to the level of danger as perceived by a person. Personal context refers to the level of comfort that a person has in divulging information about his/her movement. As a part of situational context, some situations may be perceived more dangerous than others, e.g. a sudden storm or a flash flood. As a part of personal context, some people may not care at all if their location and movement are being tracked. On the other hand, some other people may be extremely sensitive about their location and movement being tracked.

There are two important observations that motivate a multi-level security and privacy framework for CenWits. First, individual sensitivity towards tracking location and movement information varies based on the situation. Second, a lower level of security and privacy support generally translates to faster propagation of one's location and movement information to the control center. This is because a lower level of security generally incurs lower overhead. As a result, more witness records may be exchanged between hikers during an encounter. This in turn implies that the chances of locating a person in a short period of time improve if a that person uses a lower level of security and privacy. Thus, a person is more likely to accept a lower level of security and privacy, if he/she feels that there is danger. Hence, context awareness plays a crucial role in building a security and privacy framework that is acceptable to a large number of users.

5 Security Framework

It is clear that a single security model cannot address the needs of everyone under all situations. A multi-level security framework is an appropriate framework for a search and rescue system. The basic idea is that the system provides several different levels of security and privacy support, and an individual chooses an appropriate level based on his/her comfort level. Furthermore, because situations may change over time, the system also provides appropriate support that allows a user to switch from one security level to another based on the current situation.

Our preliminary design consists of five levels of security and privacy. Figure 1 illustrates these five levels along with effect of situational and personal contexts. Each of these five levels are designed to protect the location and movement information of an individual against a different type of adversary. Level 0 refers to an absence of any security or privacy support. In this level, the identity and movement information of a person is propagated without any attempt to hide it or prevent it from tampering. A user will typically opt for this level when he/she is in extreme danger. This level ofcourse has absolutely no impact on the speed at which tracking and movement information is propagated to the control center. Messages are exchanged in clear, and no sender or receiver authentication is needed. Thus, there is no security overhead.

Level 1 security provides support for protecting individual information from outsiders. At this level, hikers can exchange their location and movement information with one another with the guarantee that this information is not leaked out to an outsider. However, such a protection is not provided from the insiders, i.e. an insider (another hiker) can track the identity, location and movement of another hiker. Essentially, level 1 security has been designed to provide protection from a *passive outsider adversary.* A passive adversary is one who can simply listen to the network communication without any capability to actively insert packets, modify packets, or launch an attack against the network infrastructure. An implementation of this level requires sharing a single symmetric key among all hikers, access points and control center. All information exchanged in the system is encrypted using this key and an appropriate symmetric key protocol, e.g. AES. Level 1 requires encrypting each message before being sent. This naturally incurs some performance overhead compared to level 0.

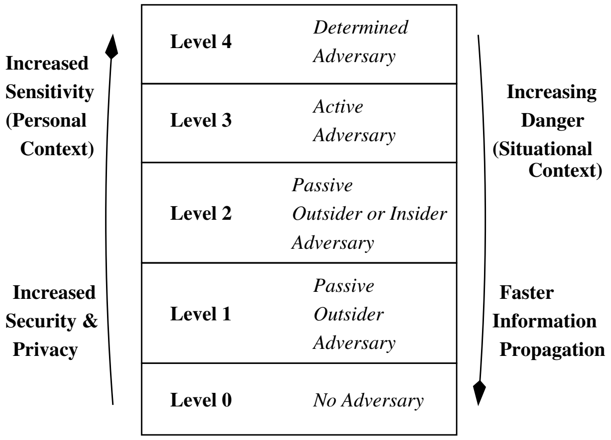
Level 4
Determined Adversary
Level 3
Active Adversary
Level 2
Passive Outsider or Insider Adversary
Level 1
Passive Outsider Adversary
Level 0
No Adversary
Increased Sensitivity (Personal Context)
Increased Security & Privacy
Increasing Danger (Situational Context)
Faster Information Propagation

Fig. 1. A Multi-Level Security & Privacy Framework.

Level 2 security provides protection from a *passive adversary.* Thus, in addition to passive outsider adversaries, level 2 security provides support for protecting individual information from passive insider adversaries as well. A passive insider adversary is one who joins the system legitimately, and records all information it can receive from the network. However, like a passive outsider adversary, a passive insider adversary does not actively seek information from other insiders, e.g. by injecting spurious packets, or modifying packets. An implementation of this level requires that each hiker share a separate symmetric with the control center. This will ensure that a witness record generated by a hiker can only be read by the control center. Additional mechanisms are needed to detect duplicate witness records. This level incurs more security overhead than level

0, because in addition to encrypting each message, mechanisms are needed to detect duplicate witness records and manage keys for each hiker.

In addition to passive adversaries, level 3 security provides support for protecting individual information from active adversaries. An active adversary is one that actively seeks individual information by targeting individuals. Such an adversary may insert spurious packets or replay packets to gain information. This adversary may be an insider or an outsider. An implementation of this level requires each hiker share a separate key with the control center. In addition, it requires incorporating mechanisms to authenticate sender's identity, determine message integrity, and detect man-in-the-middle and replay attacks. As a result, level 3 incurs more performance overhead than level 2.

Finally, level 4 security support protects the entire system from *determined adversaries*, whose goal is to degrade or destroy the complete system, e.g. by launching a denial of service attack. Such adversaries are extremely powerful and may employ very expensive and sophisticated attack methods. Defending against such adversaries is very difficult, and likely to significantly increase the cost of the entire system. Furthermore, defending against such adversaries is likely to significantly slow down the propagation of witness information to the control center. At this time, we believe that the existence of determined adversaries is extremely unlikely in a search and rescue system in a wilderness environment. Hence, we are not planning to implement this level in CenWits at this moment.

6 Discussion and Current Status

Based on this multi-level security framework, a relatively carefree individual can set his sensor to level 0 or 1 security, while a security-sensitive person can set his sensor to level 2 or 3. It is very important that a user understands the implications of setting his/her sensor to a particular level. This means that if a user sets his/her sensor to level 3, the user understands that his/her location and movement information will move relatively slowly to the control center, and that means a search and rescue team will have relatively less accurate information about his/her location in case of emergency. Similarly, if a user sets his/her sensor to level 0, the user understands that his/her location and movement information can be tracked by relatively less sophisticated individuals. However, a search and rescue team will have a more accurate information about his/her location in case of emergency.

A user can change his/her security level setting at any time. So, if a user perceives that the current situation has turned dangerous at any time, e.g. if a heavy rainfall starts, he/she may switch to a lower sensor security level to ensure that his/her location and movement information is propagated faster. An interesting question is if this switching of security levels can be automated. If the system detects that the environment has become dangerous, e.g. there has been a severe thunderstorm warning, or an avalanche has occurred, it may automatically lower the security level chosen by a user. This capability is especially important in emergency situations when a person may not be in a position to manually switch his/her sensor security level. This will require equipping the sensor device with appropriate sensing capabilities that can sense such dangerous situations. At present, we are in the final stages of completing our design of this multi-level security framework for CenWits based on these ideas.

References

1. 802.11-based tracking system. *http://www.pangonetworks.com/locator.htm.*
2. Brent geese 2002. *http://www.wwt.org.uk/brent/.*
3. The onstar system. *http://www.onstar.com.*
4. Personal locator beacons with GPS receiver and satellite transmitter. *http://www.aeromedix.com/.*
5. Personal tracking using GPS and GSM system. *http://www.ulocate.com/trimtrac.html.*
6. Rf based kid tracking system. *http://www.ion-kids.com/.*
7. F. Alessio. Performance measurements with motes technology. *MSWiM'04*, 2004.
8. P. Bahl and V. N. Padmanabhan. RADAR: An in-building RF-based user location and tracking system. *IEEE Infocom*, 2000.
9. K. Fall. A delay-tolerant network architecture for challenged internets. In *SIGCOMM*, 2003.
10. L. Gu and J. Stankovic. Radio triggered wake-up capability for sensor networks. In *Real-Time Applications Symposium*, 2004.
11. J. Hightower and G. Borriello. Location systems for ubiquitous computing. *IEEE Computer*, 2001.
12. W. Jaskowski, K. Jedrzejek, B. Nyczkowski, and S. Skowronek. Lifetch life saving system. *CSIDC*, 2004.
13. P. Juang, H. Oki, Y. Wang, M. Martonosi, L. Peh, and D. Rubenstein. Energy-efficient computing for wildlife tracking: design tradeoffs and early experiences with ZebraNet. In *ASPLOS*, 2002.
14. K. Kansal and M. Srivastava. Energy harvesting aware power management. In *Wireless Sensor Networks: A Systems Perspective*, 2005.
15. G. J. Pottie and W. J. Kaiser. Embedding the internet: wireless integrated network sensors. *Communications of the ACM*, 43(5), May 2000.
16. S. Roundy, P. K. Wright, and J. Rabaey. A study of low-level vibrations as a power source for wireless sensor networks. *Computer Communications*, 26(11), 2003.
17. C. Savarese, J. M. Rabaey, and J. Beutel. Locationing in distributed ad-hoc wireless sensor networks. *ICASSP*, 2001.
18. V. Shnayder, M. Hempstead, B. Chen, G. Allen, and M. Welsh. Simulating the power consumption of large-scale sensor network applications. In *Sensys*, 2004.
19. R. Want and A. Hopper. Active badges and personal interactive computing objects. *IEEE Transactions of Consumer Electronics*, 1992.
20. M. Welsh and G. Mainland. Programming sensor networks using abstract regions. *NSDI '04*, 2004.
21. J.-H. Huang, S. Amjad, and S. Mishra. CenWits: A Sensor-Based Loosely Coupled Search and Rescue System Using Witnesses. In *SenSys'05*, San Diego, CA, November 2005.