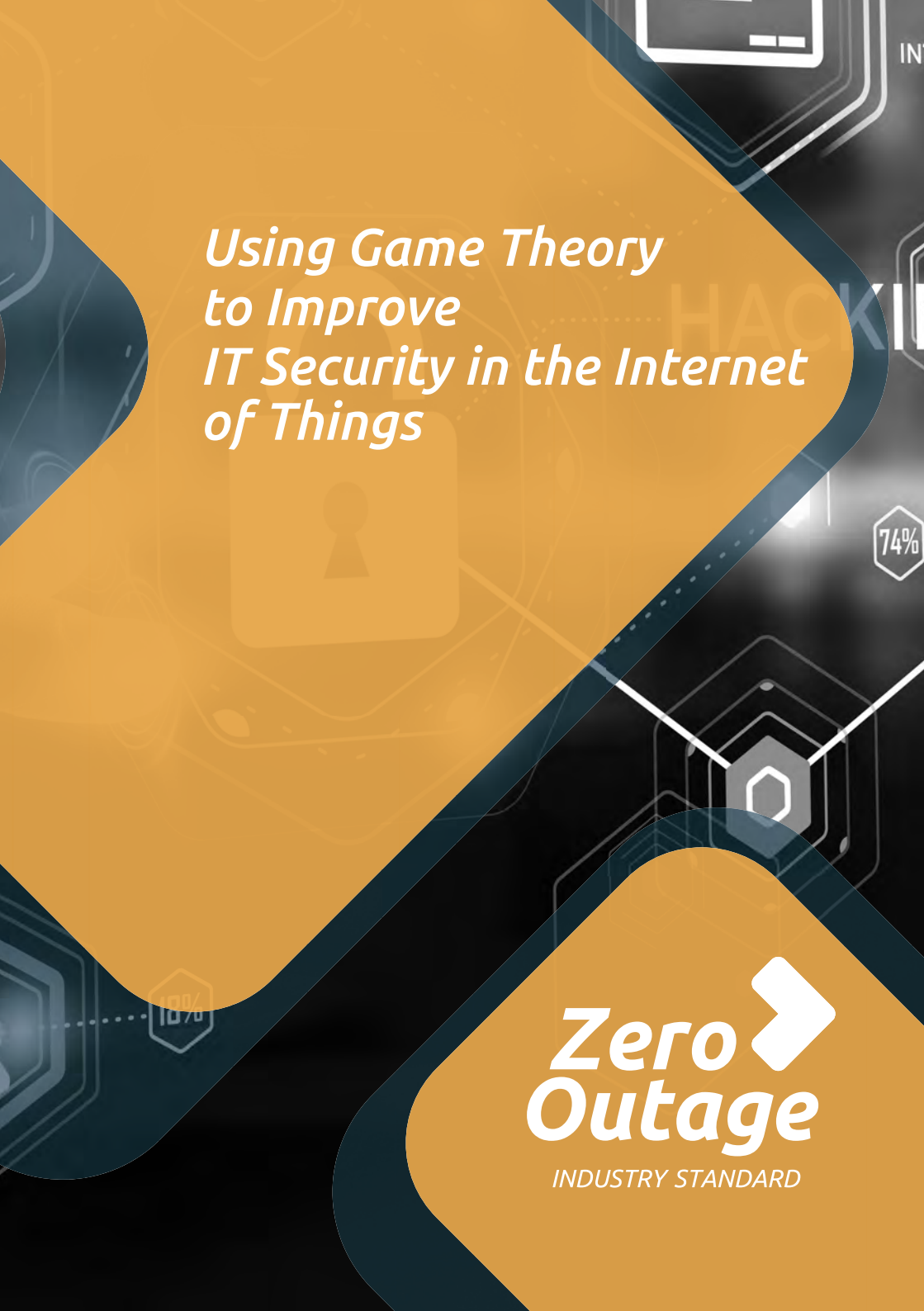
Using Game Theory to Improve IT Security in the Internet of Things



Zero Outage

INDUSTRY STANDARD

CONTENT

Zero Outage
INDUSTRY STANDARD

3 The Idea of a Durability Date or: What happens if nobody cares?

4 Introduction

5 Background and Problem Description

10 Approach to Central Computing

17 Ideas for Secure Devices in the Internet of Things

25 Discussion and Outlook

30 Literature

The Idea of a Durability Date or: What happens if nobody cares ?

Game theory is a branch of mathematics and economics. Its aim is to predict rational and hence actually observed human decisions, in order to understand the reasons for preferring one alternative over another. The decisions of market participants and/or players considerably influences the level of information security that can be achieved. The level to which security is valued depends on how important it is perceived to be and the potential that market participants have to influence one another. In the Internet of Things, there is a great deal to be found at sixes and sevens. Vulnerabilities in cameras and other devices can be exploited to turn them into weapons. Can Game Theory be used to inflict a change for the better?

Prof. Dr Eberhard von Faber

T-Systems, Chief Security Advisor, IT Division;
working areas: security architecture, developer of ESARIS, secure IT production, secure IT outsourcing, integration into processes and ITIL, standardisation, cloud, IAM

E.von-Faber@t-system.com

Walter Sedlacek

Mag., MSc MBA PMP; T-Systems; different roles in intern. Management and project management; management of the intern. Roll-out of ESARIS; head of the regional data centre in Singapore

info@waltersedlacek.com

Introduction

Computer systems, as well as other electronic, IT-equipped devices and systems frequently possess vulnerabilities (security gaps), which are able to be exploited by attackers. A common method for eliminating such vulnerabilities entails the updating of software, typically referred to as patching. This is especially true of internet-enabled devices containing software, where the initial danger is that it is first unknown whether the capability for software updating has been planned for and secondly whether it has actually been carried out. Thirdly, to make things worse, device and software manufacturers often use software from third-parties who do not feel a sense of responsibility. This article will address these issues and propose possible solutions. Additionally, it will explore how the updating of software for different computer systems can be ensured. Primarily, this process takes place in industrially operated data centres. The solution presented is to be understood as an idea and taken as food for thought. The advantages and disadvantages involved will be a subject for discussion. Scientific work on Game Theory was granted the Nobel prize for economics eight times. This emphasises the importance of this sometimes underestimated discipline and which for this paper serves as the model when applied to IT security.

Eberhard von Faber,
Walter Sedlacek

1. Background and Problem Description

1.1 Subject

Today's modern Internet unites two characteristics or trends: the de-central, distributed use and compilation of information on one hand; and the central provision of IT services on the other. Both areas are not separate from one another. On the contrary, central applications increasingly use and process data generated by de-centrally distributed components and devices; they also make data available for those components and devices. Sensors, which are distributed throughout the Internet, compile information which is centrally processed. Actuators receive their control command from central IT applications. For several years now, the number of electronic, IT equipped devices which act as sensors and actuators and also process information themselves have risen tremendously.

As such, devices are built into everyday items and in industrial plants etc., which has led to the term the Internet of Things (IoT).

We define the de-centrally distributed components and devices in a simplified manner as being devices with the Internet of Things (IoT) or abbreviated to IoT devices. (That this is a simplification, because, e.g. PCs in private hands are actually not IoT devices, is irrelevant for the discussion that follows.) Central components and devices represent the opposite.

Nevertheless, the difference between these types of devices is not really a question of where they are. It is of much more significance if they are clearly in the possession/care of an IT service provider, as these are installed in their data centre, or if the possession or the care is not so clearly regulated or recognisable.

Why is the difference so important? One may be of the assumption that an IT service provider sees the updating of the software of their systems as their job, insofar as the IT service provider is directly or indirectly affected if possible vulnerabilities are not eliminated. (This point will be elaborated upon in Chapter 2.) On the other hand, there are, however, IoT devices that do not make it clear to the user or operator if the software updating is supported and carried out. (A possible solution is outlined in Chapter 3.) First of all, the problem should be thoroughly explained and with the help of examples, be fully clarified.

1.2 Problem Description Based on Examples

Let us observe a few examples of such IoT devices to better understand this problem. First, to directly attack IT services or servers or to bring malware into circulation, attackers build up so-called bot networks. They consist of a multitude of captured computer systems, which are remotely steered without the authorisation of the owner and they are misused by the operator of the botnet.

For a long time, vulnerable PCs have been seized and made a part of such a bot network. For some time now, a new level of quality has emerged through the Internet of Things. It was reported in 2016 that bot networks such as “Mirai” convert inexpensive Internet-cameras in the hundreds of thousands for the purpose of being able to misuse them. This was made possible due to vulnerabilities in the software of these cameras.

"It is of much more significance if they [IoT devices] are clearly in the possession/care of an IT-service provider..."

Secondly, it is not so long ago that an unknown IT security specialist with the pseudonym "Kenzo2017" issued a warning that certain routers (these are not IoT devices in the true sense of the word) which households and companies were connected to, were susceptible to being remotely steered and to being exploited for attacks. The manufacturer issued a software update, but nothing happened. Thirdly, industrial plants such as wind-powered, hydro-electric plants or other machines operated by small enterprises transfer measuring data and diagnostic data over the Internet to central applications in data centres and receive control commands on the same path.

For the exchange of this data, standard components are installed in the industrial plants. The owners and the operators of industrial plants, as well as those of video recorders and TVs, are often totally unaware that these IoT devices have to be run and maintained according to security guidelines. They are part of a function of the plant, which serves an industry and business-specific purpose which the owner and the operator of the industrial plant are primarily responsible for.

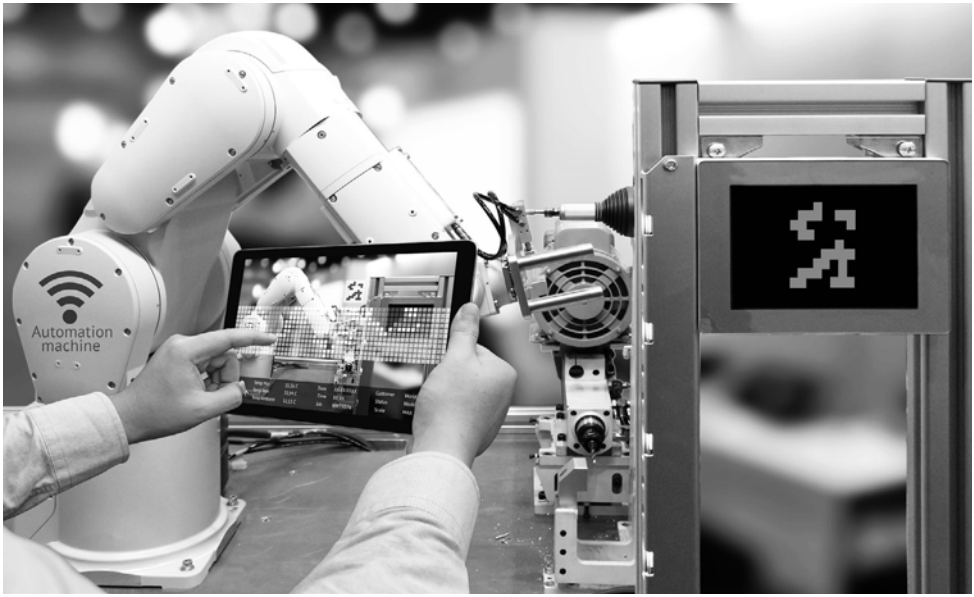
The (known) manufacturer installs a part made by a different (perhaps even unknown) manufacturer–which means, in the end, no one feels concerned with or has a sense of responsibility for the IT security maintenance required.

All three examples have something in common – the purchasers, owners and operators of the IoT devices are often unclear about the significance of IT security. Why? Purchasers, owners, and operators are often not informed in a way that enables them to immediately recognise different security levels and to be sufficiently aware of possible implications. There are no logos and no labels that show which devices and systems differ in terms of better or worse IT security. The specialists themselves also have difficulty detecting this difference.

IoT DEVICE SECURITY

In the second example, it also depends if the already provided patches (software-updating) have really been applied. The misuse of IoT devices, e.g., for bot networks can only then be adequately made more difficult when the software updating is carried out extensively and practically with all affected devices being updated.

The third example shows that there must be consequences if hidden IoT devices are not updated. Only in this case would the manufacturer of the plant have to point this out to the owner and the operator. And only then can it be expected that the owner and the operator would make the updating of the software a component of the contract a requirement and thus enforce it. In this article, the outlined solution for a specific category of IoT devices is oriented towards these three observations. However, it should be noted that for the elimination of the vulnerabilities, the updating of the software is not a complete cure: it is not a guarantee for the sufficient protection of systems, but an important prerequisite for this, because IT systems, as a rule, are not perfectly secure in the sense of being free of vulnerabilities.



2. Approach to Central Computing

Before a solution is developed for IoT devices, we should first look at "conventional" IT systems to examine how such challenges are addressed there. More precisely how this can be done with industrialised IT production, as the latter can be assumed to have the highest level of maturity. Readers not interested exploring this in further detail can skip this chapter.

We will first define what is understood by industrialised IT production. The surface area of a data centre approaches the size of a football field and houses some 2,800 racks with a total of nearly 40,000 physical servers (computer systems). Climate control, power supply and the like are not included in this figure. Such a number of systems calls for a large-scale data centre. The IT systems are run by an IT service provider, who makes their IT services available to their customers. When the customer is a large enterprise, the corresponding IT provider will be especially complex. Large-scale enterprises have very specific business processes. The support of such business practices through modern IT calls for certain requirements and solutions, which raise the complexity of the IT and TC of the IT service provider. To assure quality and to keep costs under control, the provision of the IT service is process-oriented and highly organised in a shared-task

"Similar to automobiles of today no longer being manufactured by a team of specialists, but by people trained only to carry out certain, simple tasks along the assembly line—the IT of this day and age is also industrially produced."

manner. Similar to automobiles of today no longer being manufactured by a team of specialists, but by people trained only to carry out certain, simple tasks along the assembly line—the IT of this day and age is also industrially produced.

ESARIS and the ESARIS Security Taxonomy [9] enable that the IT security be able to have command of such an IT production which is process-oriented and characterised by a high degree of division of labour. In the process, familiar measures are integrated. Hence, it has less to do with implementing measures such as access protection, encryption, monitoring, etc. (For this purpose, there are ample other sources such as [6] and [7]). On the contrary, a challenge is faced in possessing a method which ensures that hundreds and thousands of such security measures be defined, communicated and appropriately fully applied in an industrialised production environment with thousands and sometimes, tens of thousands of highly specialised employees in several countries around the globe [9].

Because the IT production environment is organised as a process-orientated manner, the ESARIS Security Taxonomy focuses approximately half of its activities on the development, implementation and the operation of IT services, including their care, maintenance and further development. As the IT production environment takes stock of a great number of technologies, the other half of the Taxonomy concentrates on the typical technology areas which also supports the division of labour within the IT service provider and with all of its partners and suppliers. At this point, only the activities in the development, implementation and the operation of IT services are of interest since—this is all about the enforcement of IT security in the broadest sense. Its characterisation is mainly derived from established procedures, as defined in ITIL. In ITIL, however, as with ISO/IEC 20000 and other similar standards, IT security on the one hand, and the specifications of an industrialised IT production on the other, are either insufficiently taken into account or not at all. These are the reasons for the augmenting of existing best practices by the association Zero Outage Industry Standard on the basis of ESARIS. [11]

Figure 1 shows a section from the ESARIS Security Taxonomy with references to processes (from ITIL) and how they are established in an industrialised IT production process. It is notable that four areas of the ESARIS Security Taxonomy have no equivalent in ITIL. This underscores why a direct taking on one of the ITIL processes would not only have been too confusing but also insufficient.

Three of these newly included areas touch upon the subject of this article directly: the Vulnerability Management and Mitigation Planning (VAM), the Patch Management (SPM) and the area of Hardening, Provisioning and Maintenance (HPM). The Vulnerability Management is so essential for the subject of "security" that an additional area (VAM) had to be created.

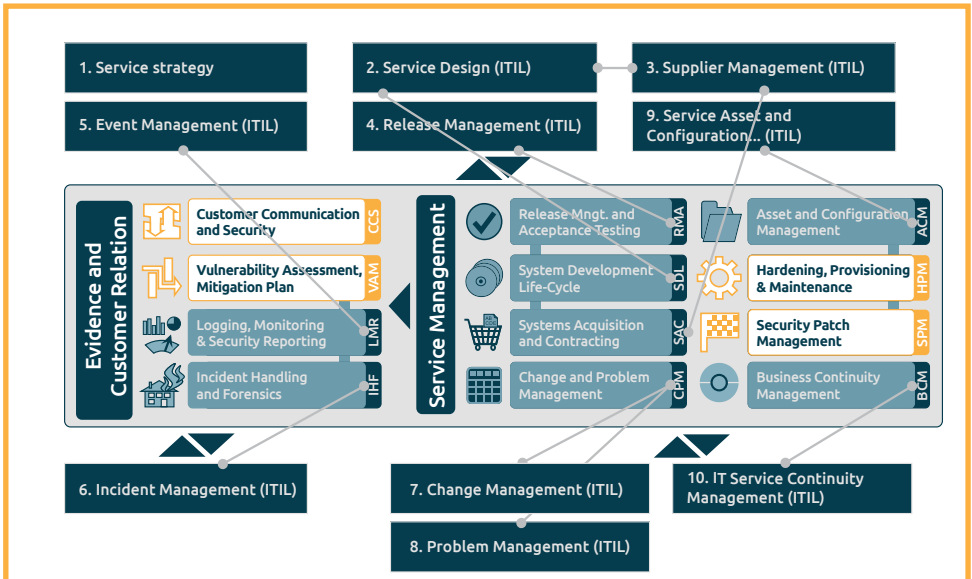


Figure 1 | Section from the ESARIS Security Taxonomy (12 of 31 areas)

A further area that was added entails Patch Management (SPM). Surprisingly, one only finds this key activity with difficulty in ISO/IEC 20000 or ITIL.[1] SPM, however, is so important that it has to be precisely elaborated on in order that vulnerabilities be systematically eliminated and security gaps are sealed.

The Hardening, Provisioning and Maintenance (HPM) consist of significant, practical guidelines for the realisation of guidelines in a general life-cycle as described in ITIL processes.

Through the shaping (and within the elaboration of the ITIL processes through ESARIS) a complete picture of a life-cycle unfolds which thoroughly models and takes into consideration IT security. The process and the connections are only outlined briefly. For further details, please refer to the cited literature.

What is particularly relevant for the subject of this article involves the following procedures (see Figure 1). The Life-Cycle Management provides the regular updating of the software (Patch). The basis for this comes from defined processes and procedural methods which include the definition of patch classes (Security Patch Management, SPM). The patch process starts with activities such as the collection, quality assessment, packaging and the scheduling for the implementation, followed by compatibility tests.

1 IT Infrastructure Library; ITIL is a compilation of proven procedures for the implementation of an IT-Service-Management (ITSM) that meanwhile represents a de facto standard. The corresponding international standard for the ITSM is ISO/IEC 20000 [8].

This implementation is managed by Change Management (Change and Problem Management, CPM). This process ensures the contingency planning and preparation of a roll-back, for risk mitigation and review and approval.
In the end, the patches are implemented and the change is reported as accomplished. The patch status is maintained and known within the Asset Management (Asset and Configuration Management, ACM).

Several security patches follow this path because the transition between function and security is often fluid. However, there is a reason for the existence of the mentioned second area of Vulnerability Management (Vulnerability Management and Mitigation Planning, VAM). For example, scanners are operated which examine the systems (in a targeted manner) as to whether vulnerabilities exist. There are several, sometimes different sources that inform on the vulnerabilities and penetration tests and forensic examinations provide further indications of vulnerabilities no matter of type.

The Vulnerability Management only performs an analysis and assessment and provides proposals for a further course of action (termed 'mitigation planning'). What comes after? If there is evidence of an imminent danger, the issue is forwarded to Incident-Management (IHF), even when the cause and the measure for remediation are already known because, for example, a patch already exists. The implementation is carried out again through the Change Process,[2] with the Patch Management Process being the final executive arm.

2 Typical "changes" are often compiled and planned in the form of a "release".
All "changes" (normal and emergency changes) are implemented through the corresponding process.

"...there is a problem for business customers, who operate the IT devices and systems which they have no idea... or are only partly familiar with."

If there is no imminent threat, the question is posed as to whether the solution for the elimination of the vulnerability is known. For instance, if a software update is available. However, the Vulnerability Management also observes other cases. In the first issue, all information is forwarded to Patch Management; in the second, the solution may not be known at all, in which case it is transferred to Problem Management.

In this way, a large-scale industrialised IT producer can ensure those vulnerabilities are systematically detected and are eliminated via the updating of software. The IT service provider can only then provide this service at the required quality level when it is a component of its own IT service. For example, if the customer allows their own applications to be executed on the infrastructure of the IT service provider, the customer can then be responsible for updating the application software, while the IT service provider only takes care of the updates of the infrastructure components.

However, it is also common for the IT service provider, possibly by the order of a third party, to assume the responsibility of updating devices and systems which are not in their possession and are also not even installed in a data centre.

This also applies to IoT devices, which are offered as so-called "Managed Services". Indeed, IT service providers often take control of this vehicle to continuously eliminate vulnerabilities by means of updating the software.[3]

Business customers can infer in their contracts with the IT service provider as to whether the updating is a part of the offered and contractually agreed services. If they explicitly buy IT service, they also take advantage of the required expertise to assess corresponding evidence in the form of security reports—which the IT service provider places at their disposal.

It is a different story for consumers and companies, who do not explicitly buy the IT service at all—but implicitly apply components and systems which actually (would) require the IT service to perform updates. Therefore, a problem exists for the consumers, who fail to grasp the complexity of this matter and its consequences; and there is a problem for business customers, who operate the IT devices and systems which they have no idea about or are only partly familiar with. The following outline solution is intended for both of these cases. It is to be understood as a concept.

3 As we always have a role-model perspective in mind, this naturally includes manufacturers offering updates as an IT-service.

3. Ideas for Secure Devices in the Internet of Things

3.1 Eye-Catcher and Solution Outline

The following almost encompasses the entire solution description. However, it is described in greater detail later using Game Theory.

- A train racing ahead can pose an extreme danger for those riding in it and for the environment when the train driver falls asleep or for other reasons, fails to fulfil his duties. Therefore, there is a so-called dead man's handle installed in the train: the driver has to activate this device every 30 seconds. If he fails to do so, he is alerted by sound. Should he not react, an emergency brake is triggered to prevent an unfortunate occurrence.

- Don't we need such a device for the "Internet of things"? IoT devices have their vulnerabilities. In the event that they are not eliminated by the updating of software, they can cause great damage and even become a weapon. What would happen if these devices had limited durability? Would the software be updated at this time or the durability extended? If not, the user first obtains a warning. Then, in the case where no response is given, the device fails to perform its duty or reduces its operation to prevent further danger. The regular updating by the manufacturer leads to the elimination of vulnerabilities and extends the overall durability.

- We call this "life sign control". The user recognises security with this seal/logo! It indicates: This device is always "fresh"; it is preserved and remains up-to-date – thanks to the manufacturer. This can be identified by anyone without any technical know-how whatsoever and without any control of the software versions, etc.

The user sees the logo with the durability period and is aware that the manufacturer looks after it. The result? Only the regularly maintained "living" devices remain on the Internet. For greater security. For better quality.[4]

It is important to note that this solution only solves the three problems identified in Section 1.2. This means the following: Firstly, consumers can differentiate between devices and systems of higher security and those of inferior IT security. The affixed seal/logo reflects this. Secondly, similar to the eco-seal, the logo itself is no guarantee that the promise implied is actually kept. The manufacturer or service provider, however, is legally obliged and quality controls can be made at any time. As IoT devices are mass-produced goods, the software updating is applied to a great number of devices. However, this still allows for control. In any case, this updating is carried out extensively and, as a rule, proactively, without the user having to take the initiative. If users express a preference for such devices with the sign/logo, the diffusion of the solution will be significantly increased.

4 The term "durability" was first created at a press conference of the "Chaos Computer Club" though it was defined differently: 33C3: Hackers demand a minimum durability period for Internet-linked devices; see: 33C3: Hacker rufen nach Mindesthaltbarkeitsdatum für vernetzte Geräte; Heise online, 31 Dec. 2016, 2:40 p.m.

Thirdly, while the second scenario described is, for many computer systems and consumers, a standard one, the non-importing of the updated software, as a rule, does not lead to any consequences for the user or the manufacturer/service provider. However, because the dead man's handle, following a warning, leads to a limitation in terms of functionality or to the complete de-activation—the consequences become clear. In this initial sense, it actually doesn't have anything to do with devices failing to perform their service. Much more, it is concerned with the solution that everything is being done to carry out the updating on a regular basis. According to the Game Theory, this involves the mechanism of "credible commitment" which ensures that those involved are not hindered from complying with their commitment to other, apparently more important things.

Credible commitments (also called pre-commitment or commitment devices) are “strategic moves” in a “sequential game”. Such strategic commitments are intended to change expectations and behaviour. In our case, the strategic move intends to have an effect both on the party giving the commitment (“meet it”) and more importantly, on the other market participants (“rely on the commitment”). For more detail refer to [1].

Naturally, the outlined solution is not employable for devices whose key characteristic is instant availability, e.g. life-support systems. More on the limitations are covered in Section 3.3. In the following Section 3.2, the details of the implementation are outlined.

3.2 Implementation Information

The solution should not make highly technical demands on IoT devices. The requirements which follow behave independently to the implementation of the solution and are only required to be able to support the functionality of the software updating. As with regards to mass-produced goods, this is not to be taken for granted, although even processor chip cards offer such features. Further below, the requirements are described that refer to the solution itself.

- The IoT device has to possess the capability to load software patches and to update the software. In this sense, along with the IT functionality, a correspondingly larger, writeable, non-volatile memory is also necessary that is capable of program execution. In order that interruptions, e.g. in the power supply do not lead to a defect in the IoT device, the memory has to be large enough. However, the actual functions for the realisation are known and up-to-date.

- The IoT device has to have the possibility to be able to test the authenticity of the patch before the new software version becomes active. On the size of the memory, see above. The examination of the authenticity includes the proof of the data origin and the proof of the integrity of the patch. Generally speaking, for this purpose, signatures or MACs are used, which do, however, require the capability of the IoT device to carry out cryptographic operations with algorithms of sufficient strength and to be able to manage the applied cryptographic key. In addition, the following functions are necessary for supporting the proposed solution:

- With every software setup (patch) the IoT device receives a date and consistently saves "time of the last update". The IoT device has to be in the position to determine the "current time". For this purpose, it either has an internal real-time clock or regularly establishes contact with a time server.
- The IoT device regularly defines the difference between "current time" and the "time of the last update". If one uses Unix time, it is only necessary for the determination of "elapsed time" to subtract the two 32-Bit-long numbers from one another. A second such subtraction provides the comparison of "elapsed time" with the "durability period" saved in the device.[5]
- If the "elapsed time" is greater than the saved "durability period", the IoT device must be in the position to reduce its functionality to such an extent that no danger can arise from it. If the "elapsed time" gets close to the stored "durability time", it would then be desirable for the IoT device to be able to issue a warning. This process is schematically depicted in the following figure (Figure 2). The non-technical requirements include, among others, the following:

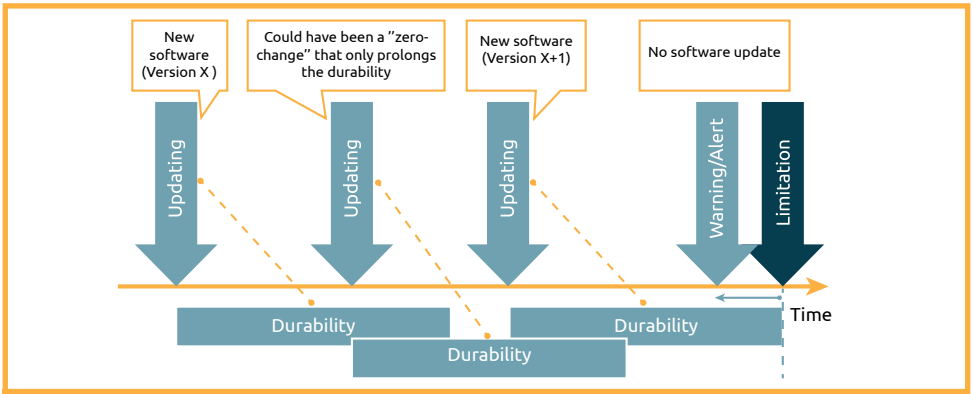


Figure 2 | Schematic representation of the function of the solution

5 Although the difference is measured to the second, it probably really results in a matter of days. The "durability period" can be weeks or even months.

- The rules for the implementation of the solution and its associated implications have to be prepared very accurately and be publicly available.
- There has to be a version of these rules which is also fully understandable to all people. The sign/logo has to possess a sufficiently recognisable and informative value.
- The sign/logo should also be made known. It signals that the party who uses this sign/logo for products or services is obliged to comply with the rules. (This also means that users and manufacturers take into account the defects of IoT devices, should the manufacturer or the IT service provider decide to no longer carry out the updating. This also applies even if it is no longer in the position owing to, for example, its business becoming insolvent.
- The sign/logo may also additionally convey the following information: type of warning (e.g. email, announcement, none) and length of the "durability period" (e.g. weeks, month, months).

The technical requirements make the credible commitment really credible and finally enforce it. In addition, there are non-technical requirements. They are often underestimated. Game Theory also addresses these. The solution of the game strongly depends on the information being available to each player. Not knowing the strategy of the other player may cause oneself to act unpredictably (e.g. to apply a so-called mixed strategy [1]). The attempt to take advantage of the non-observability of a strategy usually leads to a result that is worse for both sides [3]. Therefore, one should try to develop the game towards one where the information is complete so that the preferred result is easier to achieve, or becomes more likely.

Questions of a judicial nature are not the subject of this outline.

3.3 Limitations

It has to be pointed out that with the outlined solution there is a conflict of interest between security and availability. This is due to the fact that functional limitations occur in the event the manufacturer or the service provider fail to meet their commitment or cannot fulfil it and are unable to look after the updating of the software. This is the Achilles heel of the solution! This updating, however, is absolutely necessary, as shown below. Therefore, there will be a class of IoT devices for which this solution is not applicable! One could consider if it makes sense to offer a somewhat deviating solution for this case, the implementation of which, once again, would have to be indicated via the sign/logo.

With regards to software updating, a common misconception is that it is not required to be regularly updated. If there are no vulnerabilities which need to be eliminated by the manufacturer, there must be a "zero patch", which indicates that the IoT device is up-to-date and only the "time of the last updating" has to be updated. However, this also means that the manufacturer/service provider, within the span of the "durability period", has to create and apply at least one patch.

The solution doesn't mean that the updating of the software has to be carried out on a totally automatic basis and without the involvement of the user. Here, quite different implementations may be carried out. For IoT devices that are not serviced or maintained by the user, an automatic "push-service" is provided.

For devices in which the user directly works with the IT functionality, it can be beneficial to allow the user to delay the updating of the software for a certain period of time, e.g. if the updating is tied to a temporary limitation of the functionality or availability. This is known to be the case with privately owned Windows computers.

4. Discussion and Outlook

There are certainly more complex and more effective solutions to increase the security of IoT devices. The one introduced here stands out because it doesn't require its own central infrastructure and its implementation is oriented towards the limitations of a large number of IoT devices, like embedded systems.[6]

This is why it should be kept in mind that the solution is only designed to solve the problem initially described in this paper: With a certain class of devices, there is the risk that it is not identifiable if at first, the updating of the software is supported and secondly, if it is actually carried out.

It is based on the assumption that more complex solutions may not be implemented or only find use in a very limited area. Unlike with payment transaction systems, in which the evaluation of IT security is prescribed and approvals are the prerequisite for use, devices connected to the Internet do not represent a closed system, but an open one. Furthermore in this context, we are not dealing with a controlled system but with a free market. Therefore, it was the aim to set the requirements as low as possible. Naturally, this also means that the effect is limited. However, it is hoped that should the proposed solution ever come into operation that its contribution will still be significant.

A further assumption includes that it is not wise to assume that good solutions are finally chosen over bad ones. IT is more a "market for lemons", in which products of higher quality are driven out. This is particularly applicable in the segment of consumer products, but not only there. The "market for lemons" is a phenomenon described by George Akerlof (Nobel Prize winner for Economics 2001) arising from the assumption that consumers are less informed about the characteristics of a product (than the manufacturers) and thus tend to

6 Embedded Systems are systems (a) with a computer-functionality, which are, however, (b) not freely programmable, but built only for and supporting a very specific application-purpose, which, (c) as a rule, interact with their environment, hence exchange information, and (d) are specifically subject to structural-technical limitations. The latter-mentioned characteristic is of high importance here. IoT-devices often are mass-produced goods with a low price, small size and low operational cost being the key aspects.

buy cheaper products even if they risk picking the "lemon" instead of the "orange". Refer to [2].

The way out of the dilemma (fewer lemons and higher returns) exists classically in the clearing-up of the information-asymmetry where the customer is better informed. However, this isn't the case in IT because consumers do not have the required know-how at their disposal to understand such information and its consequences and to be able to draw the right conclusions from it. Primarily, the sign/logo informs but it can do much more! It signals (1) a guaranteed promise and allows (2) the manufacturer/service provider concerning the associated credible, communicated commitment to really take care of the updating. The credible, communicated commitment is reached through this or strengthened to the point that breaking the promise leads to serious consequences: The device is not as capable of functioning as expected and has already been paid for. This may raise the level of pressure exerted on the manufacturer/service provider enormously.[7]

"THE WAY OUT

of the dilemma ...exists classically in the clearing-up of the information-asymmetry where the customer is better informed."

Yet, what does the manufacturer/service provider get out of this? What could their motivation for subjecting themselves to these rules of credible commitment? There are a great number of manufacturers/service providers for whom the updating of the software is already included in the service. They will gladly grasp the idea of a sign/logo because it will not cause any additional effort for them. It will, on the contrary, make their improved performance more visible, because they are selling

7 It could also be considered to encourage state authorities to support this seal/logo, similar to the "Blue Angel" eco label in Germany, as both cases, in a broader sense, involve the protection of the public domain.w

sweet "oranges". This visibility puts the producer of the sour "lemons" under pressure. The question is how the user makes their decision. Will they risk buying "lemons" while they could have had "oranges"? This question cannot be answered at this stage. There are too many factors which influence both the purchase of IoT devices and the dynamics of their installation.

> ***"There are a great number of manufacturers/ service providers for whom the updating of the software is already included in the service."***

Which risk do users accept and are they motivated to favour these "new" solutions? At first sight, the labelled devices have a basic fault. They fail to operate if the manufacturer/service provider is no longer prepared to be involved. But how likely is it that the market leaders, who dominate the purchasing decisions, will visibly break their promises? The authors estimate the likelihood to be somewhat minimal. Of higher probability is the case of an insolvency or a change of the business model, which inevitably leads to the promise no longer being kept. Harm to one's reputation with repercussions for the follow-up business is not likely. At most, it could affect other areas of business.

What has to be expected from all of this is that the life of IT, as a rule, is limited and IT is subject to rapid changes with high renewal rates. In this context, it is absolutely intended, for security reasons, that obsolete, "forgotten" and no longer serviced devices be sorted out through the solution.[8]

On the other hand, users may accept a risk when buying IoT devices which have no such logo. These devices obviously have a flaw right from the start; they are neither maintained nor improved.

Moreover, the user or operator of such devices takes the risk that the devices are abused.

8 The updating could also be bound on a warranty period. This would decrease the financial risk for the manufacturer due to the necessity to maintain "second-hand equipment".

However, the proposed solution is not a universal remedy as has been previously emphasised. In particular, it is not applicable for all classes of devices. Roughly, one could put those devices or systems operating on the Internet into three classes and assign these varying solutions as follows:

- For devices or systems with a high computing capacity, the described solution in Chapter 2 of a central, contractually agreed on updating of the software is to be preferred and to be regarded as adequate.
- For devices or systems with a medium range of computing capacity, the new solution (see Chapter 3) is exactly the right one. These systems make for the largest group and fastest growing number of things in the Internet of Things. This is currently considered to be the main problem.
- For devices and systems with a very limited computing capacity or with very high availability requirements, the first version disqualifies itself, owing to a lack of technical requirements. The second version may be realisable in technical terms, but the high availability requirements forbid it from being implemented with the new proposed solution provided here.

IT is complex. It is not bad if a solution doesn't heal the world. A solution suffices when it fulfils a clearly defined purpose and keeps its promise. The present discussion paper is to be understood exactly in this context. The proposed solution of a "durability date" is to serve as a suggestion with regards to IoT devices, in order to evoke discussion about security on the Internet.

Decisions play a major role in IT security. The use of concepts of Game Theory can help to understand the decisions of participants and to shape the environment in such a way that preferred results become more likely or will even be enforced. The use of the concept of credible commitment (as a strategic move, [1]) and the conclusions drawn from the analysis of the lemon markets [2] are just examples. Information and communication play an important role. Game Theory helps to understand that these aspects are not just simple prerequisites. On the contrary, shaping information and communication can help to advance the game into one with complete with information and to obtain payoffs that make the preferred behaviour more likely [3].

The potential of Game Theory is, however, not exhausted with this. Mechanisms of coalition games [3] can help to understand the effect of agreements between the players. The study of repeated games with complex starting points (e.g. the prisoners' dilemma) can show how payoffs and future prospects in the form of controlled repetition and discounted pay-offs can lead to more cooperation [4]. Finally, the fundamentals of Game Theory remind us that the adequate analysis of alternatives and the estimation of benefits are very important [5]. Although the pay-off functions are usually constructed based upon assumptions and imperfect appraisals, the analysis of the correctly classified game mostly provides amazing clear and helpful strategic proposal.

Literature

[1] Avinash K. Dixit and Barry J. Nalebuff: Spieltheorie für Einsteiger, Strategisches Know-how für Gewinner; Schäffer-Poeschel Verlag für Wirtschaft Ulm, 1993; English title: Thinking Strategically; W. W. Norton & Company, 1993

[2] Ken Binmore: Fun and Games, A Text a Game Theory; D. C. Heath and Company, 1992

[3] Manfred J. Holler und Gerhard Illing: Einführung in die Spieltheorie; Springer, 7.Auflage, 2009

[4] Robert Axelrod: Die Evolution der Kooperation; R. Oldenburg Verlag, München, 1987; English title: The evolution of cooperation

[5] Morton D. Davis: Spieltheorie für Nichtmathematiker; R. Oldenburg Verlag, München, 1993; English title: Game Theory: A Nontechnical Introduction

[6] ISO/IEC 27002 – Information technology – Security techniques – Code of practice for information security management

[7] Informatiown Security Forum (ISF) – The Standard of Good Practice for Information Security 2016

[8] ISO/IEC 20000 – Information technology – Service management – Part 1: Service management system requirements, Part 2: Guidance on the application of service management systems

[9] Eberhard von Faber and Wolfgang Behnsen: Secure ICT Service Provisioning for Cloud, Mobile and Beyond, ESARIS: The Answer to the Demands of Industrialized IT Production Balancing Between Buyers and Providers, 2017, ISBN-978-3-658-16481-2, XIV + 368 pages – 2. UPDATED and EXTENDED edition

[10] Eberhard von Faber: Organisation der Absicherung einer industriellen ITProduktion, Drei Handlungsfelder jenseits von Protection, Detection, Reaction; Datenschutz und Datensicherheit (DuD), Heft 10, 2016, Seiten 647-654

[11] ESARIS Security Taxonomy – Synopsis, Scope and Content; Zero Outage Industry Standard, Release 1 about Security, February 2017, https://www.zero-outage.com/security

DuD • Datenschutz und Datensicherheit 1 | 2017 7

The information contained in this document is contributed and shared as thought leadership in order to evolve the Zero Outage Best Practices. It represents the personal view of the author and not the view of the Zero Outage Industry Standard Association.

CONTACT

Zero Outage Industry Standard Ltd.
Suite 1, 3rd Floor
11-12 St. James's Square, London
SW1Y 4LB
United Kingdom

info@zero-outage.com