Empowerment of Security Engineers through Security Chartering in Visma

MONICA IOVAN, Visma
DANIELA SOARES CRUZES, Sintef
ESPEN AGNALT JOHANSEN, Visma

A security program consists of a set of activities, projects, and initiatives to be implemented in a coordinated manner, in order to meet business objectives and realize the company's information and cyber security strategy. As the program has to work in self-managed teams, Visma has realized that a compliance-driven approach would not be the optimal solution to the security program strategy (top-down approach to security). Visma has chosen to pursue an approach where security becomes part of the teams' routine (bottom-up approach to security). Empowerment of the teams is then an important success factor. This paper presents the Security Chartering technique in Visma that is used to evaluate the effectiveness and get feedback on the program and most importantly, empower the security engineers by giving them a voice to raise their concerns, and to share success cases and experiences with the program.

1. INTRODUCTION

Software security is the idea of engineering a software system so that it keeps working correctly even under malicious attack [MCGRAW]. Over the last few years the threat landscape has changed with a continuous increase in the number of threats, advanced attacks, hackers collaborating to make freely available Open Source tools and databases with malicious intent. On the other hand, the focus on software security in the software development teams are not increasing in the same direction, and the pressure for functionalities output is still winning the battle on the prioritization of activities inside the teams. We have observed two main reasons for this. First, there is pressure from the market and from stakeholders regarding the speed of developing new features. Second, the teams do not always see the importance of these security activities [CAMACHO].

A security program consists of a set of activities, projects and initiatives to be implemented in a coordinated manner, in order to meet business objectives and realize the company's information and cyber security strategy. As the program has to work in self-managed teams, Visma has realized that a compliance-driven approach would not be the optimal solution to the security programs strategy (top-down approach to security). Visma has chosen to pursue an approach where security becomes part of the teams' routine (bottom-up approach to security). Empowerment of the teams is then an important success factor.

The proper handling of software security activities requires specialized tools and knowledge. However, agile development teams are generally small, and do not have specialists in security. One approach to overcome this challenge is to use the concept of *security champions* or *security engineers*. Security engineers are team members that are responsible to promote and support the adoption of security activities inside the team without breaking agility, continuous delivery, self-management, and autonomy. The security engineers are valuable for the security initiative because they understand better the challenges and cultures of their teams and can better adapt the directives from the security program to ways that the team can relate to and are able to act on them and adopt as part of the activities of the team on a daily basis. They also collect information about the specific risks of the products and manage them. For the security engineers' initiative to work there is a need for a set of activities to support this role. This includes onboarding, regular meetings, training, provision of tools etc., to ensure they have the tools they need, and the power to decide and act based on their knowledge.

Another important aspect to consider in the security program is that, with almost 300 teams, validating the effectiveness and efficiency of the program is not an easy task. To evaluate the effectiveness of the program, it is important to understand the effects of the program on the self-managed software development teams.

Monica Iovan, Timisoara, Romania; email: monica.iovan@visma.com
Daniela Soares Cruzes, Trondheim, Norway; email: daniela.s.cruzes@sintef.no
Espen Agnalt Johansen, Oslo, Norway; email: espen.johansen@visma.com

Amongst the approaches used in Visma to validate the program, get feedback and empower the teams, Visma has started to use a technique called Security Chartering. In this paper, we present this technique for empowerment of the Security engineers. It is a specific focus group approach that helps us understand the effectiveness of our product security strategy, get more focused feedback on the program, provide an additional channel of communication from the teams to the security core team, and understand the priorities of the teams towards improvements needed for the program. Most importantly, Security Chartering provides a good channel to empower the security engineers by giving them a voice to raise their concerns, and to share success cases and experiences with the program. This also helps us to validate the hypotheses we had about the security program, giving us the confidence that it is working. An additional advantage is we obtained a list of improvements for the security program itself.

2. SECURITY AT VISMA

2.1 Company Context

Visma is an international software company, headquartered in Norway with a presence across the entire Nordic region, as well as Benelux and Central and Eastern Europe. Visma delivers software that simplifies and digitizes core business processes in the private and public sector. Acquisitions are also essential to Visma's strategy. The Visma group is currently a federation of around 145 individual companies with over 11,000 employees. These independent companies share infrastructure and services.

Due to this diversity, each organization is composed of self-managed teams and each team is responsible for the entire lifecycle of their service, including the security of the service. Depending on team composition, it is possible that not all teams have personnel with security expertise, and this could lead to poor security for some products.

In order to ensure that we deliver all company's products with a high security standard a centralized team was created, named Product Security Team, and having the role to facilitate the security for all Visma's products. Their scope is to create the core of the security program by providing initiatives that make sure all products are secure, and at the same time keeping the self-management of security inside the individual teams. The Product Security Team is built around an ambidextrous Security Program in a way that the program uses both top-down and bottom-up approaches [CRUZES]. In order to ensure the bottom-up approach the teams selected at least one security engineer that will empower the rest of the team in making security decisions.

Once the security engineers have been chosen, the Product Security Team guide them through a Security Self-Assessment form, which is an extensive list of questions about different aspects regarding security of the product. As there are many concepts that are not familiar to the developers, after answering this questionnaire, they realize they have acquired a substantial amount of knowledge during this process. They are also invited to connect with other security engineers through a special online *Guild* meeting and a slack channel. This way they are informed about news in the security and also are briefly informed about relevant security events in the company and outside the company.

2.2 Software security initiative challenges

Having over 300 self-managed teams, each with its own way of working, and a governance-led culture where centralized Product Security Team is driving the software security efforts using only a top-down approach had the benefit of creating a standardized way of working across teams. However, it was not effective in achieving changes of behavior uniformly across and within the software development teams. The challenge was to find ways to complement the top-down approach with a bottom-up approach where also development and operations teams drive software security efforts. With a bottom-up approach, the attention to security becomes embedded as part of the team culture, establishing in this way what we came to refer to as the *ambidextrous* security program.

The ambidextrous security program has many benefits. Some teams are innovation-driven in their way of working; these teams are more willing to prioritize security tasks, to focus on automation or prototyping different security controls. For these teams the bottom-up approach fits their needs, and the challenge is to ensure the balance of this innovative approach with governance measures in order to make sure that all security components are considered. Some other teams are overwhelmed with their own way of working, are not open to change, and are more inclined to a top-down approach. This top-down approach takes the form of

controls around assurance, such as policies, standards, and gates. The biggest challenge with these teams is balance promoting a security culture with empowering the team members to take security decisions.

After almost three years running this program, we have observed that the teams vary in motivation to work on security tasks. Sometimes the variation was on the ability to act on identified security vulnerabilities because of the pressure to quickly deliver different features, or due to personal motivation of the security engineer. Even though the Product Security team recommends having volunteers as security engineers, for some teams, the team managers were forced to appoint one of the team members for the role. Appointed security engineers were in general less interested or have less knowledge about security, and the onboarding process can become overwhelming for them.

Our observations lead us to conclude that a top-down approach can be better for short-term achievements since forcing the teams can generate fast results. While a bottom-up approach requires a longer time and new ways of thinking, and since not every activity is right for every team the bottom-up approach helps to customize the security activities to the team needs. If the teams do not balance the two approaches well it is easy to forget one of them and to either go for a governance-led approach or to ignore some security aspects and choose only the ones that are more attractive to work with.

In order to make the bottom-up approach work we need to make sure the employees are empowered to do the security activities. There are two methodologies of empowerment. The first is about *external influence* that comes from management or the organization. This methodology focuses on employees' possibility to suggest, and to get involved, or on the effects of positive affirmation from someone with authority. The second one, called *psychological empowerment*, focuses more on the perception of the employees, and is about the belief that they have necessary knowledge and skills to perform the job well and can make a difference in the organization [GEORGE].

Many of these aspects require good communication between the organizational members. In order to have good communication there is need to have a two-way communication mechanism where employees can raise their voices and where they can get feedback for their actions, in other words to have an effective feedback loop.

2.3 Mechanisms for feedback

Merriam-Webster's dictionary defines feedback as the transmission of evaluative or corrective information about an action, event, or process to the original or controlling source. The first step is to identify the feedback sources. Each source can provide a different perspective to the security program.

The most used sources for feedback are surveys such as NPS, KPIs and real-time data. Surveys are an objective way of receiving feedback but can also be misleading because no metric is perfect. For example, a high NPS value for training can indicate the quality of the training is good but does not identify if the quantity is sufficient. Another source of feedback are the members of the Product Security team since they have a better understanding of the delivery teams' performance than other supervisors or upper management. However, since they are driving the program, it could be, that they become subjective to the challenges the teams have implementing the program. We also noticed that many times the improvement points became scattered and it was not easy to have an overview of all the aspects of the program that needed improvement. Managers typically are another rich source of feedback. They are experienced and have specialized knowledge of the tasks their teams are performing. They also have insight into company procedures, policy, and roadmap prioritization. However, the most important source of feedback is still the delivery teams. They are the ones that implement all security activities and they are the ones who understand the pain and gain of these activities.

In Visma, we decided to use different methods (surveys, meetings with the team members, communication channels and customer focus groups) to collect different perspectives of the security program. Each mechanism has advantages and disadvantages. E.g., in the case of written feedback even if it is a fast method to collect feedback it has the disadvantage that the feedback is taken out of context and it can create misunderstandings. In one scenario, a negative feedback may be interpreted by the receiver as a reprimand, and then this might reduce receiver's motivation to act on the feedback.

The most broadly used feedback mechanisms in our Security Program are security guild meetings [Smite], different slack channels, retrospective sessions, different feedback forms, and NPS scores. Some of these are

described further in Table 1. All these tools complement each other. However, on seeking to have more insights into the hearts and minds of the delivery teams, we felt the need for two-way feedback communication and at the same time to increase their sense of belonging in the extended Visma security team. This is why we created the *Security Chartering meetings*. These meetings are feedback collection sessions from the teams on the security program in Visma. It is a structured focus group meeting, where the teams discuss the implementation of the security activities in their teams by addressing:

- the confidence of security in the team;
- the impressions on the security activities that are suggested in Visma in the security development life cycle, asking them what they would like to keep, add, do less and do more;
- the motivation to do security activities, discussing what motivates and demotivates them to work in the security activities in a daily routine.

Characteristics	Security Guild	Slack Channel	Security Chartering
Organizer	Product Security Team	Product Security Team	Product Security Team
Participants	Security Engineers	Security Engineers All employees	Security Engineers Developers Service Owners
Structure	Centralized, virtual	Centralized, virtual	Clustered by locations
Periodicity	Biweekly meetings	Slack channels	Annual
Value for the members and the company	Access to expertise Forum for expanding skills and expertise	Network for keeping abreast of a field	Sense of belonging Two-way communication
Challenges	Low engagement Size and distribution One-way communication	Low engagement Size and distribution Insufficient activity	Gaining the trust

Table 1. Characteristics of different Feedback Mechanisms used in Visma

Some elements of empowerment relevant to the Security Chartering include:

- Skills and resources – sharing information and being transparent
- Power to decide and act – having authority, opportunity and motivation
- Rewards system – being responsible and accountable for outcomes of their actions

Comparing retrospectives with security chartering we concluded that security chartering is a cross-team activity and not inside a specific team, giving more sharing, helping the participants to relate to each other, and giving them the possibility of learning from each other. Seeing what other security engineers are doing also helps deepen understanding of what their role is.

3. SECURITY CHARTERING SESSIONS IN VISMA

We ran the Security Chartering from July to October 2019, with a total of 9 sessions (2 hours each) and 78 participants. In each location, we had two sessions, one for Security Engineers and one for Developers. The first step was to analyze the teams in order to see where they are located and what is their security maturity level according to the Visma KPIs. We selected teams from our four biggest clusters, where Visma has a higher concentration of security engineers. Then for each team we selected the security engineer and a developer. We selected a mix of experience levels and gender. We also tested an online version of the security chartering by selecting several security engineers that were in more remote locations. We facilitated the meetings to support the participants to discuss the topics they considered important. The results of these sessions were aggregated and presented to the Product Security team, in order to evaluate which actions needed to be taken and what should be prioritized. The results were then presented back to the participants, by showing the projects and the tasks created based on the action points. We encouraged them to volunteer to work on these projects. Two outcomes of the Security Chartering sessions are:

- a feedback loop that will be used to facilitate easier adoption of the security activities;
- a document that contains the needs of the development teams and prioritization of the security projects based on their needs. Below is a summary of the main topics that appeared in the sessions.

3.1 Training and Knowledge Sharing

In Visma, the Product Security team organizes annual, one-day on-site trainings, in different locations for both non-technical personnel (business analysts and managers) and for developers. In the morning, the developers training is mostly focused on the OWASP (Open Web Application Security Project) Top 10 vulnerabilities [OWASP] and in the afternoon, they participate in a *Capture the Flag* [CTF] exercise in order to apply what they previously learned. These trainings are highly appreciated by the employees in Visma, with high scores on the internal evaluation. In addition, by design all services in the Security Program have a central element of providing training in its core. For instance, in the Static Application Security Testing (SAST) service, we have learned that most developers need some time in that service before they learn to do secure coding. The same element can be seen in the other services and one might argue that for the developers the entire Security Program is a training program. Although not always explicitly addressed as such.

Trainings needs were the main topic of discussion in all locations. The main feedback from the participants was that they would like to have deeper learning on security. Participants mentioned there is a need to add more types of security-related training, for different roles, for different subjects and in different formats. We also noticed the need to have more sharing of previous experiences with security breaches between the teams. Based on these requests the Product Security team started to make changes to the training offering for this year, looking to offer an online training tool, different awareness campaigns, creating possibilities for the teams to share their security experience with others, organizing hack the box events and many others.

3.2 Prioritization

In Visma, security engineers have in general 20% of their working time allocated for security activities. Examples of these security activities include self-assessments, integrating security services into the build pipeline, and validating the cases that come from bug bounty or from the internal penetration tests. Teams have at least one security engineer, but some bigger teams have also decided to add more security engineers to the team because of the size and complexity of their products. For example, they have more integrations and the self-assessment questionnaires and approval, or the threat modelling requires more time and knowledge compared to other products.

It is important to note Visma has a security maturity index system. We measure the security maturity level using a penalty points system, which has four levels. When a team is not following the activities designed to given security activity, they receive a penalty, and this affects their actual security maturity level. Through the security maturity index, we identified that some of the teams prioritize security activities while others are constantly delaying these activities.

When we asked how much time they allocate for security and how security work is prioritized only some of the security engineers mentioned they have enough time to work on security activities. The developers mentioned that all team members should be more involved in security activities, and therefore the teams could allocate time for that in the sprints.

We noticed that in some cases the security engineers and/or the managers are empowered to have all the security activities prioritized. When needed they even use more than the allocated time for the tasks. These teams are the ones achieving a higher security maturity level score. Due to a big backlog or lack of empowerment, other teams are not using the allocated time, and this is visible in their security maturity level. Based on these findings we created another Security Chartering session for the service owners in order to understand better the difference in their behavior towards prioritization of security activities. We also suggested to the developers and security engineers to take charge and to be security leaders in their teams. They should help the managers understand the risks and they should start working on the security activities.

3.3 Security Engineer Role

Even though the Product Security team recommends having volunteers as security engineers, for some teams this was impossible, and in these cases, the team managers appointed one of the team members to the role, based on seniority or security experience. It was more evident in the Security Chartering sessions that the security engineer role can be at the same time both motivating and demotivating, depending on the way they were chosen for this role (appointed or volunteered) and on the way they were onboarded.

It was clear in the security chartering that the appointed security engineers needed a more focused onboarding on the role. While the volunteers started working on the security activities in a more organic way. We then decided to put more focus on the onboarding process of security engineers.

We also noticed there is a need for a clearer understanding of their role, and also the need of an onboarding process to the security engineer role. In response, we agreed to create an onboarding package for participants in the security program. At the time of writing, the creation of this package is in progress, and will contain information about the program and their role, a plan for trainings and invitations to all the channels used for communication. The benefits of using this onboarding package will be observed in the next couple of years.

3.4 Whole Team approach for Security

We noticed a high degree of variation of the understanding of how the responsibilities of the security activities are distributed inside the teams. Some of the participants believed that security engineers were responsible for implementing all security activities, while others believed that security engineers are the facilitators of these activities and all team members have the responsibility to contribute. Based on this, we have decided to address the whole team approach to security explicitly in the onboarding of the teams to the security program.

4. LESSONS LEARNED

We categorized the lessons learned on the experience of running the Security Chartering sessions as improvements to security program and pitfalls and opportunities, as described below.

4.1 Improvements to the Security program

The overall impression that the participants of the security chartering gave is that they are satisfied with the resources they receive from the security program; that they have the basic trainings, and they learn more by doing the security activities. Still they would like more coaching and more advanced trainings. This is a good sign for the security program since it can be interpreted that the security engineers are fast evolving and they want to go to the next level of knowledge.

As described before, another aspect of empowerment is the decision power. From the Security Chartering sessions, we can conclude that most of them showed to have the freedom to decide what is best for their service security, and some of them feel empowered to volunteer to work on the security program. On the other side, we noticed that they would like more support from their managers in order to prioritize faster the security tasks.

Sense of belonging is as an important aspect of the empowerment of the security engineer. Our conclusion was that the security engineers did not recognize themselves as participants of the security working force in Visma. Some of them only felt like they were executants of the policies that were made by the Product Security team, indicating that the Product Security team needs to empower them to use the bottom-up approach more. Following the self-management approach, if they don't feel empowered, they will only stay in this level of self-management, and they might lose the possibility to influence the security program in Visma, even for their own benefits and needs. Therefore, after the sessions, the Product Security team started to prepare a much closer collaboration and we have decided to focus more on having a two-way communication channel where transparency is the most important aspect. Using this system, it is possible to be transparent on what projects are the Product Security team working on and they can get fast feedback for their actions, encouraging this way the collaboration between them.

We can conclude that these sessions are a good mechanism to validate that the ambidextrous security program in Visma works well. The participants gave feedback that the security activities are useful, interesting and important. In general, the participants did not want to remove any of the security activities from the program, even when these activities were time-consuming or overwhelming in periods for the team.

4.2 Pitfalls and Opportunities in Running the Security Chartering Sessions

During the preparation of the Security Chartering sessions we noticed that in locations where one of the organizers was known to the teams and we had already established trust with software development teams, there was a much higher interest in participating to the sessions. Therefore, for the other locations we used a local "figure" that helped us in promoting the event. Where needed, to ensure adoption and less friction the manager of Product Security team sent emails to participants to ensure they understand that the research is

legitimate and had the support of upper management. On the invitations we sent we explained what and why we were running the sessions, but we did not provide an agenda. This intrigued the participants and some of them admitted that they participated in the sessions because they were curious to see what will happen while some others initially declined the invite due to the fact that did not understand what is expected from them. In the end, we concluded that they did not understand what to expect from these sessions, therefore, in the future they would like to have a better description in the invite in order to give them time to prepare.

By not having a list of topics to discuss, we let the participants voice the subjects that are important for them. That is one of the reasons we ended up discussing different topics in different sessions. We noticed that it is important to have a balance between levels of experience of the participants due to the fact that they will come up with different concerns. Beginners in the security engineer role focused more on trainings and onboarding, while the security engineers that were in the role longer focused more on advanced topics, such as improving manual security testing. However, having all levels of experience raises challenges to the session facilitation because we need to make sure all participants have the possibility to raise their voice and concerns without feeling restricted.

It was also interesting to notice the differences in culture in different locations. Some locations were more inclined to a top-down approach, doing what the management "allows" them to do. In these locations, the participants asked their managers for permission to participate to these sessions and even some managers asked for more details before they approved the participation. This is showing us that empowerment is not yet universally achieved, and this enforces the need of having different activities to increase empowerment.

Team structure also had an impact on the how the security activities are coordinated inside the team. For example, in some small teams, one developer has multiple different roles and security engineer is just one more role that the person has in the team. In contrast, some bigger teams decided to have multiple security engineers and to split the work between them and so they were able to better manage the prioritization of tasks. But even so, many of the small teams are more empowered to prioritize the tasks.

One of the sessions was run online. The participation of the security engineers in the online sessions was not the same. Engagement of participants was much weaker. We tried to reproduce the same types of interactions that we would have in the in-person sessions, but it did not work the same. Participants lost the opportunity to see others from their own sites in person and create a more personal connection to the colleagues.

One of the challenges we faced when we planned these sessions was to understand how many chartering sessions we needed in order to still have benefits. In other words, how to evaluate the saturation of the findings. We decided to stop when we saw that the topics coming from these chartering sessions were more than the Product Security team was able to absorb. Our plan is to run the same sessions in two years, prioritizing the locations and the teams that did not participate before.

Another challenge we faced was translating the findings to actionable changes in the security program. To perform these changes, we had to persuade the Product Security team that the security-related findings were deemed relevant and useful. One of the barriers to the findings from the Security Chartering was that the team members mentioned that some of the findings were already known to them, but they have not yet prioritized the changes to address the challenges mentioned by the participants. It is important that the team members do not see these findings as criticism to their work, but as an opportunity to meet teams' needs for improvements.

5. CONCLUSIONS AND FUTURE WORK

Empowering employees can help the organization to improve the adoption of security activities within the self-managed software development teams. Our goal is to empower the teams to become pillars in the protection of the organization's systems and data, therefore protecting customer data and Visma's reputation. Security engineers play an important role in this empowerment of the teams. Visma has implemented an ambidextrous holistic approach to the security program that is both top-down and bottom-up.

The Security Chartering sessions helped us to holistically and systematically identify the needs of team members involved in the product security activities, empowering them and ensuring that the bottom-up approach is not forgotten. In addition, these sessions offered us a more objective way of validating our security program. The Security Chartering also helped us to identify the needs for adaptation in the services that the security team provides to the delivery teams to improve the adoption of the activities. Through the Security

Chartering, we created a communication arena. This arena gave team members the opportunity to raise their voice, articulate their concerns, and discuss success cases and experiences with the program. Overall it helped establish a common understanding of the security engineer role, its purpose, and context.

Furthermore, the sessions created other effects that were not foreseen by us initially. There was a significant effect on the trust of the development teams in the transparency and openness of the security team in Visma. We noticed an increased amount of communication about sensitive cases regarding security inside of the teams and an increased willingness to ask for help in some specific areas. Developers and security engineers started to address us directly with feedback about the program. We noticed an increased commitment and ownership to the security program. In addition, we also perceived an increase of motivation to start adopting some security activities that they have not started before.

Based on the action points we elicited from the security chartering sessions, as future work, we will continue focusing on empowering the security engineers even more by supporting the growth of an open security community in Visma. We plan to open our security backlog so that all team members will have the possibility to contribute to the security program. In addition, the feedback loop needs to be promoted in order to benefit as much as possible from the two-way communication and in order to embed the security culture in the organization. We also plan to implement a rewards system, where for example we will have a multi-level security engineer role. Prioritization of security activities is an open challenge, and we are now defining approaches to have a more systematic way of handling priorities for security activities across the different development teams. For that, we will have to work on a top-down approach by ensuring managers are prioritizing the security activities, and also on a bottom-up approach by empowering the team members to influence the decisions on priorities based on the evaluation of security risks.

6. ACKNOWLEDGEMENTS

We are grateful to Ken Power for helpful comments and guidance on this experience report. We also thank the Visma and all the participants of the Security Chartering sessions. We also would like to thank the Research Council of Norway for the research grants on the SoS-Agile project: Science of Security in Agile Software Development (grant number 247678).

REFERENCES

[MCGRAW] Gary McGraw, 2004. Software Security. IEEE Security & Privacy 2(2), 80-83.

[CAMACHO] Cristina Rosa Camacho, Sabrina Marczak, Daniela S. Cruzes, 2016. Agile Team Members Perceptions on Non-functional Testing: Influencing Factors from an Empirical Study. ARES 2016: 582-589

[CRUZES] Daniela S. Cruzes and Espen A. Johansen, 2020. Building an Ambidextrous Software Security Initiative, to appear in Balancing Agile and Disciplined Engineering and Management Approaches for IT Services and Software Products, eds Manuel Mora, Jorge Marx Gómez, Rory O'Connor, Alena Buchalcevova, IGI Global 2020.

[SMITE] Darja Smite, Nils B. Moe, Georgiana Levinta & Marcin Floryan, 2019. Spotify Guilds: How to Succeed With Knowledge Sharing in Large-Scale Agile Organizations. IEEE Software. 36. 51-57. 10.1109/MS.2018.2886178.

[OWASP] https://owasp.org/www-project-top-ten/

[CTF] https://owasp.org/www-project-security-shepherd/

[GEORGE] Elizabeth George, Zakkariya K.A., 2018. Psychological Empowerment and Job Satisfaction in the Banking Sector. Springer International Publishing_Palgrave Pivot