



Co3で支えるペパボのセ キュリティ対策

~Communication Completely Continuous~

GMOペパボ セキュリティ対策室

Agenda

- セキュリティ対策室とは
- エンジニアリングで組織を動かす
- セキュリティ対策を継続させるプロダクトの開発
- まとめ



Section 1

セキュリティ対策室とは

組織と活動背景の紹介

Tamon Kumano

ミッション

GMOペパボのセキュリティ対策室とは

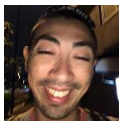
情報セキュリティ基本方針を遵守

お客様、お取引先様、従業員から預る情報資産を
適切に扱える文化形成、技術的仕組みをリードする組織

セキュリティ対策室 (2021年現在)



CTO,CISO: 栗林 @kentaro



マネジメント: 熊野 @mod_osho



エンジニア: 伊藤 @hiboma



エンジニア: 森田 @mrtc0



技術顧問: 力武先生 @jj1bdx

名前

熊野多聞 くまのたもん



経歴

- 2007年入社～
- インフラエンジニアとして各サービスのインフラを担当(キッティング～構築、運用、保守)
- 2015年～
- OpenStackを用いたプライベートクラウド基盤移行を契機にマネジメントへピポット
- 2018年セキュリティマネジメントの推進

その他

- 趣味はロードバイク
- 平日はZWIFT、天気がよければ横須賀、三浦半島、平塚あたりを走ってます





Section 1

セキュリティ対策をやりきれ る組織 ~completely~

ペパボが運用するサービスとセキュリティ対策室の関係性

Completly

対策を "完了" させること。完了させなければ、穴が空いたまま。

- 脆弱性の修正を完了させる
- インシデントを収束させる
- セキュリティ対応を完了させる

対策を終わらせることで、次の課題が見つかる => 多層防御のマインドを持つ

複数の事業部門とそれぞれのチームで開発・運用

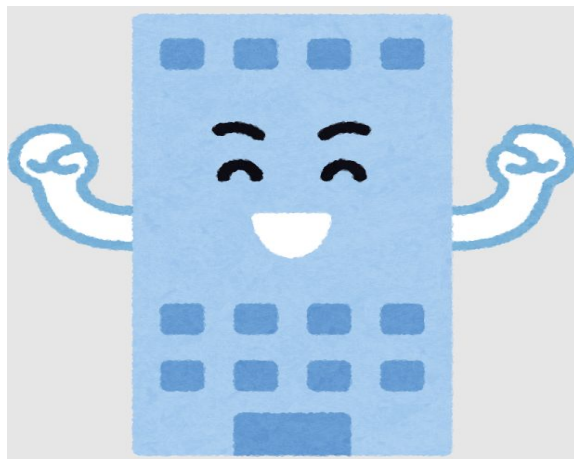
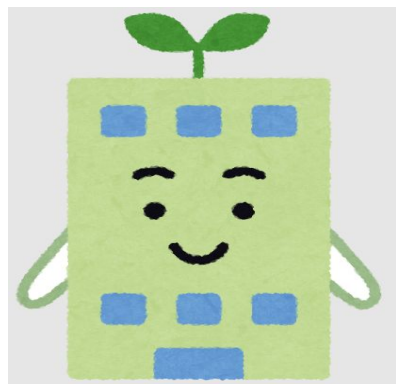


複数部門のサービスから成り立つセキュリティ事情

ペパポには10年以上続くサービスからリリースして数年まで規模や関わる人々に濃淡がある

- サービス設計時のセキュリティ対策も年月とともに変化
- 少しずつ改善することで頑健性のあるサービスになることもあれば年々メンテナンスが難しくなることも
- サービス開発時期によっては技術スタックに選択の幅がある
 - e.g. 開発言語、フレームワーク、OSディストリビューション、MW

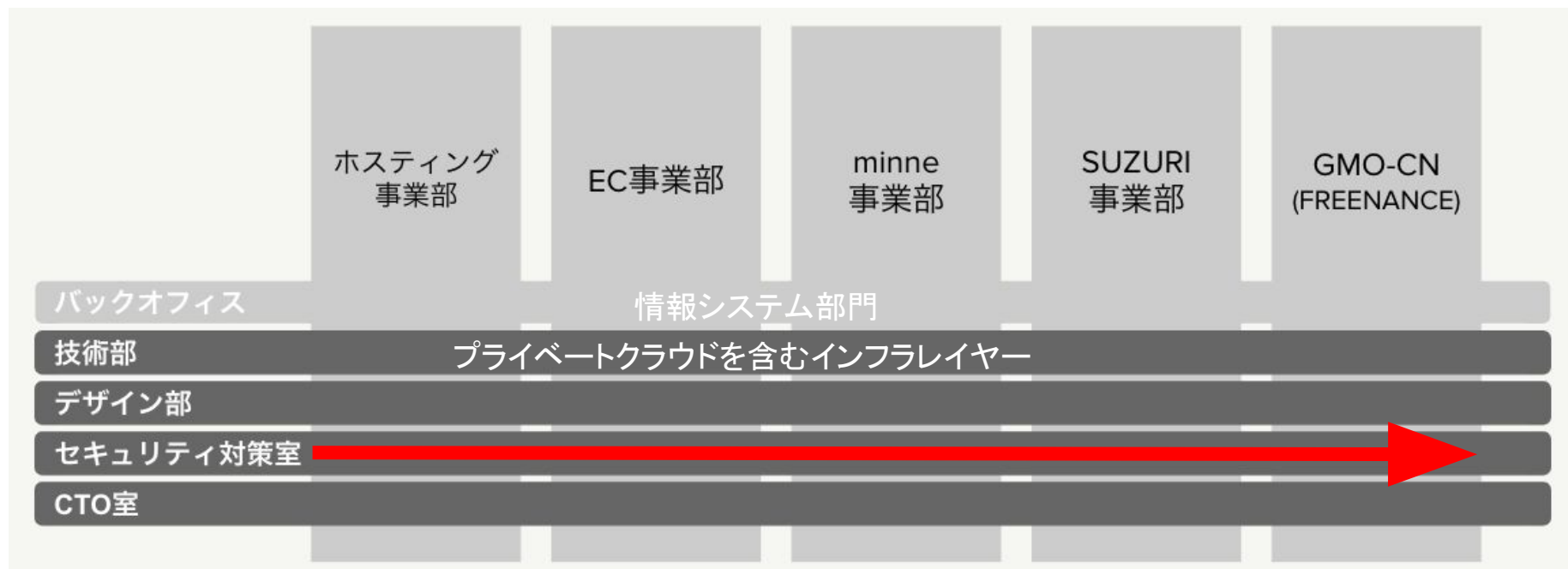
若いサービスから長年続くサービスまでさまざま



セキュリティ対策室設置前

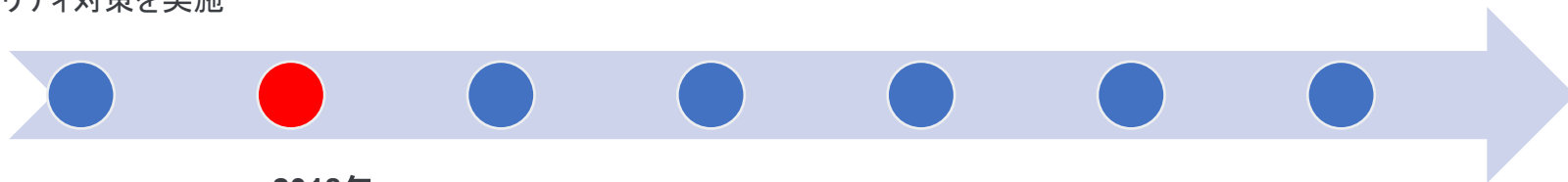


セキュリティ対策室の位置づけ



ペパボのセキュリティ対策年表

各事業部門がアプリケーション/インフラチームがOS・MWのセキュリティ対策を実施



2018年
ターニングポイントとなるセキュリティインシデント発生

ペパボのセキュリティ対策年表

各事業部門がアプリケーション/インフラチームがOS・MWのセキュリティ対策を実施

2018年3月
セキュリティ対策室発足

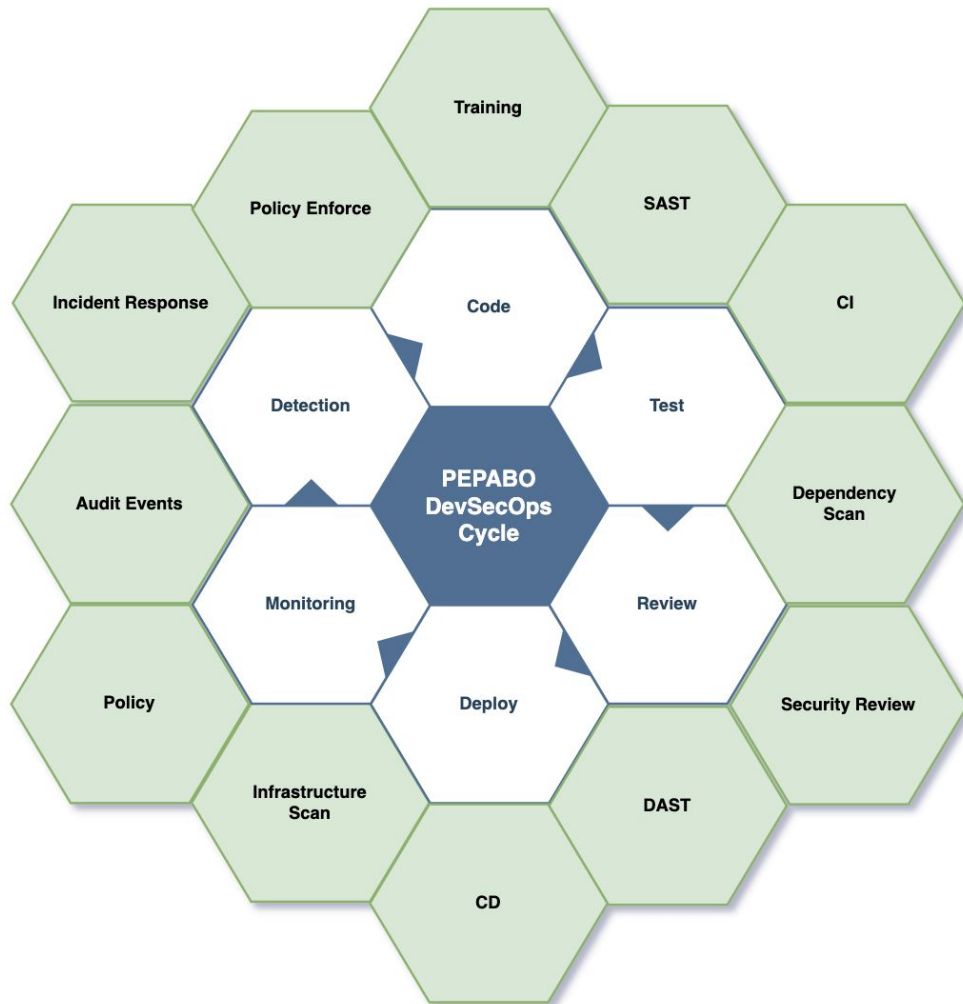
2019年
インシデントハンドリング手法の標準化

2021年～

2018年
ターニングポイントとなるセキュリティインシデント発生

2018年12月セキュリティインシデントの再発防止策完了

2020年
DevSecOpsを実現するための仕組み、ツール開発



セキュリティ対策室が取り組む DevOpsとSecの関係性

- セキュリティ対策の地図を作る(マッピング)
- 道路を舗装(ツール等環境の準備)
- ガードレールで予防(予期せぬ事故でも最悪のケースを防ぐ)

セキュリティ対策組織を運用しての発見と課題

- 外部からの攻撃も内部事故によるセキュリティインシデントも弱いところから起こる
- セキュリティのベースライン、ガイドラインを作ることで弱いところの発見と対策が実施できる
- セキュリティ対策は担当部門に任せず、協力し、やりきる、継続的に行う
- インシデントの発見者には感謝を
- 小さなセキュリティインシデントでも共有を行うマインド
 - 現場のリーダーや経営管理者によるセキュリティ方針の同期

セキュリティ対策の方針

- **Co3 (Communication / Completeness / Continuous)**
 - **Communication** ... コミュニケーションを取る。「みんなと仲良くする」。
 - **Completeness** ... インシデントを収束させる。対策を完結させる。
 - **Continuous** ... 対策を継続する。継続にレバレッジする技術を使う。
- 「ガードを下げたらやられる」というマインドを持つ
- Security as a Code で道の舗装やガードレールを作る

ペパボのセキュリティ文化を一
緒に形成してくれるメンバー大
募集！！1



Section 2

エンジニアリングで Communication を回す

Hiroya Ito 伊藤洋也 / @hiboma

自己紹介

- **名前**
 - 伊藤洋也 (いとうひろや) @hiboma
- **経歴 / よくやってること**
 - 2007年入社～
 - ホスティングサービスの開発
 - サーバサイドの基盤 API 開発
 - Linux のトラブルシューティング
- **私生活**
 - 2020年9月 栃木県那須塩原市に移住 リモートワーク!



「セキュリティ」と私個人のキャリア

2018年3月 セキュリティ対策チームに異動となった
セキュリティについては **熱心** には取り組んではないキャリアだった

- 「セキュリティ」あんまり自信ない
 - 徳丸本は読んだ
 - XSS, CSRF, SQL インジェクションくらいは知ってるが...

さて、どうやって **ペパボのセキュリティ対策チーム** をやっていこう？

チーム初期の課題

新設されたチームには課題が山積み！エンジニアリングだけでない！

- **チーム内コミュニケーションを回す**
 - 新卒で入社した森田 @mrctc0 とどうやって走るか？
- **組織内コミュニケーションを回す**
 - 少ないメンバーで組織横断的にレバレッジをかけるには？
- **エンジニアリングでコミュニケーションを解決する**
 - slack bot でコミュニケーション支援する



Section 2

チーム内コミュニケーション

チームといいつつ二人三脚

メンタリング

伊藤 @hiboma は 2007年から在籍で社内の内情を知っていた。
入社したての 森田 @mrtc0 のメンターとしてふるまう

- ペパボの技術スタックをチュートリアルする
- 抱えている問題・課題を共有する
- 同僚間のコミュニケーションを繋ぐ
- 上位等級エンジニアとしての背中をみせる

セキュリティの経験・技術をすでにもつ若手を自走できるまで
支援することもチーム初期の課題だった

シニアエンジニアに昇格までを支援できたのも嬉しい



リバースメンタリング

逆に、セキュリティのいろはを森田 @mrct0 から教わる(今も)

- セキュリティ対策の基本
 - 種々の攻撃方法や防御緩和の方法
 - 脆弱性のみつけかた
 - 脆弱性の報告の仕方
- セキュリティの技術や各種ツール
- セキュリティ業界・コミュニティ・イベント情報



チーム発足初期に双方が教えをリードできて、補い合えたことは好遇だった



Section 2

組織内コミュニケーション

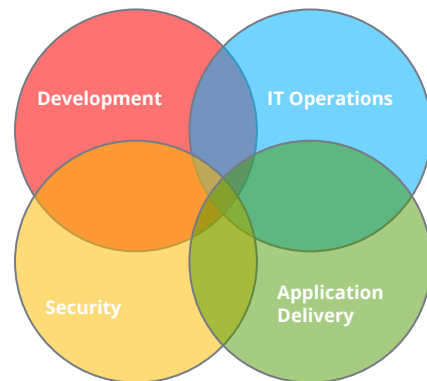
小さなチームを同僚の力を得てレバレッジする

Communication

「DevOps」vs「Sec」の対立にならないようにする

"コミュニケーション"を通して DevSecOps モデルの理解を得てもらう

- セキュリティを組織文化として育む
- 小さな異変や気づきを気軽に共有できる組織
- 平時・緊急を問わず 迅速・正確な対応につなげる

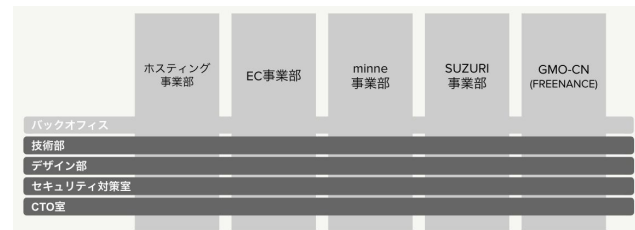


コミュニケーションが「セキュリティを当たり前とする文化」を育てる "レバレッジ" となる

小さなチームと大きな会社

社内の情報をつぶさに追いかけていくのは困難。
周囲から情報を **push** してもらえるとありがたい ... !

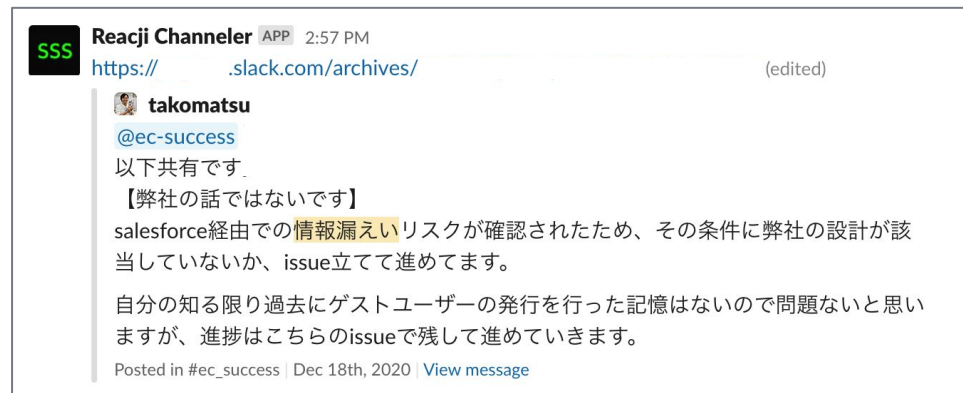
- 例: ユーザさんのお問い合わせで 疑わしい内容があった
- 例: 脆弱性の修正方法で迷っていたので相談したい
- 例: インシデントの疑いのある事象が起きた
- 例: 世間で話題になっているセキュリティのネタがある
- ... etc




気軽に情報を共有してもらうには ちょっとコツがいる

Slack で情報共有と集約を工夫

Reacji Channeler を使って簡略に実現する。  emoji を付けるとセキュリティチャンネルに共有される



熊野が全社員向けの情報セキュリティ研修に  のチュートリアルを加えたことで
技術職だけでなく、CS 職からの共有も多くなり助かっている(不審なお問い合わせの共有など)

社内イベント(ペパボテックフライデー)で宣伝と啓蒙

毎月行われる技術者向けの社内イベントペパボテックフライデーでチームの宣伝や対策の啓蒙を行う



エンジニアがリーダーシップをとろう

@hiboma の持論

インシデントにおいて

技術職がリーダーシップをとる責務を負うべき

(例外: 情報セキュリティインシデントでは CS 職のケースが多い)



sss emoji も何度も宣伝を行って普及を狙った。

新設のチームのPRを繰り返し、組織内でのポジションを確立しコミュニケーションを回していく



Section 3

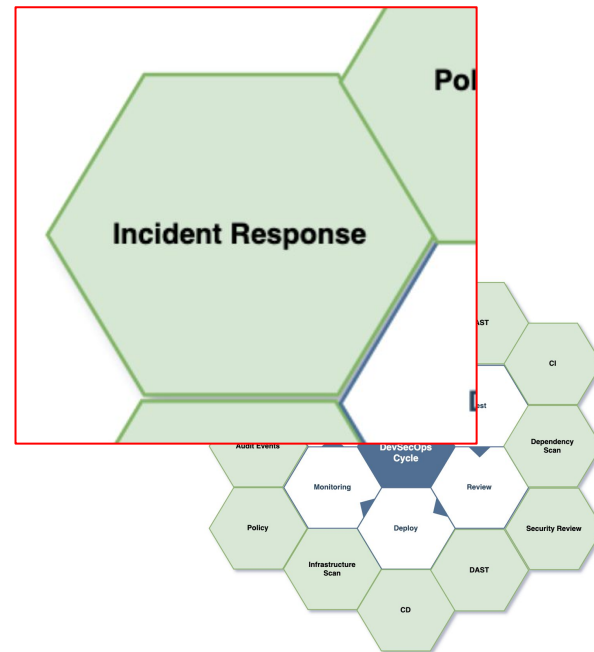
エンジニアリングで コミュニケーションを解決

slack bot でコミュニケーション支援

レジリエンス強化の課題

DevSecOps + シフトレフトの考えで、前倒した対策が
森田やサービス所属のエンジニアによって整えられていく

一方で **レジリエンス** = 障害やセキュリティインシデントが
起きた場合に迅速に回復する体制も整えていく必要もあります。



インシデント対応のコミュニケーションロスが復旧・緩和に響く

- インシデント対応でエンジニアは極度のプレッシャーに晒される
 - 未知の事象・不確実な状況も復旧を確実に達成しなければならない
- チャットツールの混乱
 - アラートを受診するチャンネルは大量のメッセージで溢れ やりとりが追いかけていく
 - メッセージが多いチャンネルで Slack のスレッド機能を使うと情報が局所化する

インシデント対応のコミュニケーションロスが復旧・緩和に響く

- 職種間やサービス・部署間の情報共有が滞る
 - エンジニアは復旧作業で詳細な技術情報(ログや分析)を書くが、そのままでは他職種には解釈が難しい
 - マネージャー職・CS 職へ状況伝達がうまくいかないケースがある
 - インシデントが他サービスに伝播した場合も、連絡が滞ると影響範囲が拡大する
- 組織内のコミュニケーションのロスで復旧・ユーザー対応が遅れてしまう
 - ビジネスへのネガティブインパクト 🔥

=> プレッシャーのかかる状況下でコミュニケーションの支援が必要

slack bot で インシデント対応を支援する

緊急時のコミュニケーションを自動で支援する slack bot = sssbot を開発し、解決に臨んだ

- インシデント対応チャンネルの作成
- 初動対応のチームをinvite する
- インシデントの発生を複数のチャンネルに通知する
- インシデントの issue を作成し記録する
- 15分後に時間を知らせるタイムキーパーを行う
- postmortem の pull request を自動作成する

... たくさん!



チャンネルを作る

チャンネル名

チャンネル名は最大80文字までです

サービス

いまいまの状況

対応の温度感

[Learn more about sssbot](#) Cancel 作成!

チャンネル作成のフォーム

チャンネルはインシデント対応ごとに都度作成する
記録一元化し、事象を追いかけてやすくする

複数のチャンネルに通知を飛ばし 事象発生を共有する


 **sssbot-staging** アプリ 16:25

 障害対応チャンネルが作成されました

チャンネル	サービス
<code>#example-20210224i</code>	other
サマリ	温度感
DB の master で障害	 RED すでにヤバイ. 緊急の対応をする

- エンジニアチャンネル、マネージャー職チャンネル、CS 職チャンネルにも通知がとぶ

初動対応チームを invite して初動を促す



sssbot-staging アプリ 11:58
インシデントレスポンスチームのアクションを確認しましょう

- 1 インシデントハンドラー をアサインしましょう
- 2 インシデント判断フローチャート に沿ってインシデントレスポンスを進めましょう

サービスの対応マニュアルも参照しましょう

- 初動対応を整える指示を出します

初動対応チームに状況を伝える



sssbot-staging アプリ 12:04

@hiboma チャンネルを作ってくれてありがとうございます. いまいまどんな状況でしょうか?

- 影響範囲は?(どんな機能や画面で影響でてる?)
- ユーザー対応の要否は?(お知らせ、お問い合わせ、ソーシャル対応は必要?)
- 原因・復旧の目処はつきそうか?

- インシデント対応チャンネルをつくった人に、状況説明を促します
- 初動チームは突然 invite されて、状況を理解できないことがあります

タイムキーパーが介入して状況判断を促す



The screenshot shows a Slack notification from a bot named 'タイムキーパーくん' (Timekeeper-kun). The notification includes a status icon with 'SSS', a clock icon, and the text 'チャンネルの作成から 2時間 経過しています' (2 hours have passed since channel creation). A 'コマンド' (Command) dropdown menu is visible on the right. Below the notification, the handler is '@ryu', the event level is '1', the detection status is '?', and the initial time is '2021-01-06 15:08:41 +0900'.

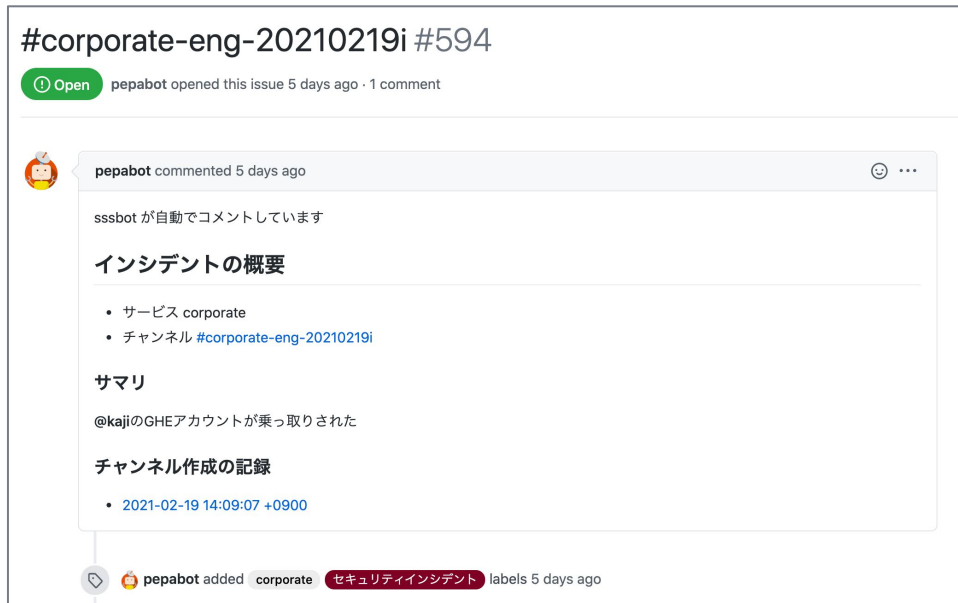
タイムキーパーくん APP 5:08 PM
チャンネルの作成から 2時間 経過しています

コマンド

ハンドラー: @ryu
事象レベル: 1
検知: ?
初動: 2021-01-06 15:08:41 +0900

- 15分ごとにタイムキーパーが介入します

GitHub issue に自動で記録をとる



⚠️ スクリーンショットは、弊社で定期的 to 実施しているインシデント対応訓練での利用例です

postmortem の pull request も自動でつくる

2021/02/19 corporate #corporate-eng-20210219i セキュリティインシデントのポストモーテム #599

1 Open pepabot wants to merge 1 commit into master from corporate/corporate-eng-20210219i

Conversation 0 Commits 1 Checks 0 Files changed 1 +57 -0

pepabot commented 1 minute ago

sssbot が postmoretm の雛形を自動生成しました

概要

- サービス corporate
- slack チャンネル #corporate-eng-20210219i

サマリ

(編集してください)

関連のリンク

- Close #594

Reviewers

No reviews—at least 1 approving review is required.
Still in progress? Convert to draft

Assignees

No one—assign yourself

Labels

- corporate セキュリティインシデント
- 緊急レベル1

Projects

None yet

Milestone

No milestone

Linked issues

Successfully merging this pull request may close these issues.

#corporate-eng-20210219i

注意事項:

- Pull Request のタイトル、概要、サマリをインシデントに即した内容に変更しましょう
- Pull Request のタイトルにチャンネル名を残しておきましょう。 redirect がレビューコメントや git push 等のイベントをインシデントチャンネルに通知してくれます

みんなの、ふつう、あたりまえを徐々に変えていく

一年ほどかけて徐々に bot の実装をつくりあげ利用を普及させていきました。

コミュニケーション支援 = 同僚のふるまいを変える対策は継続的かつ粘り強いフォローが必要です


- 全てのインシデント対応チャンネルにセキュリティ対策チームも加わる
 - CTO, VPoE も必ず invite される
- sssbot の不満点やフィードバックを拾い上げ、修正・改善・機能追加を漸進的に行う
- sssbot の機能や活用事例を社内イベント(ペパボテックフライデー)で繰り返し宣伝する

組織文化 = みんなが、ふつうに、あたりまえに、sssbot 使ってくれるまでやる!!

コミュニケーション支援から復旧・解決のリードへ

インシデントの技術的な問題は「人間」が解決しなければいけません。

SRE ともオーバーラップするように、セキュリティ対策チームはインシデントの復旧・解決にも関わります。



Technology, Engineering, Creative, and Human-Centered Design

2020-06-26

ペパボ トラブルシュート伝 - TCP: out of memory -- consider tuning tcp_mem の dmesg から辿る 詳解 Linux net.ipv4.tcp_mem

トラブルシューティング

ツイート いいね1 | シェア B!ブックマーク136 Pocket 49

セキュリティ対策室の伊藤洋也です @hiboma

過去の障害対応中に遭遇した TCP: out of memory -- consider tuning tcp_mem という dmesg を端緒として、Linux カーネルがどのようにTCPのメモリを管理するのかを調べました。

最新文章

2021-02-22
ペパボSREケーススタディ - ロリポップ！レンタルサーバーのSLI/SLOをもとにしたパフォーマンス改善の取り組みを紹介します

2021-02-10
ペパボのログ活用基盤『Bigfoot』を使った Zendesk のデータ可視化

2021-02-09
ペパボが気になる人への FAQ

2021-02-05
ステークホルダーマップの活用



Technology, Engineering, Creative, and Human-Centered Design

2020-06-11

ペパボ トラブルシュート伝 - node プロセスの general protection fault を追う - abort(3) の意外な実装

トラブルシューティング

ツイート いいね1 | シェア B!ブックマーク42 Pocket 43

セキュリティ対策室の伊藤洋也 @hiboma です。

業務中に、Haconiwa コンテナで動くある node プロセスが **general protection fault (一般保護違反?)** を起こして dmesg にログを残す現象を調べ、問題解決にあたっては、その際の痕跡をまともなおして記したエントリーになります。

最新文章

2021-02-22
ペパボSREケーススタディ - ロリポップ！レンタルサーバーのSLI/SLOをもとにしたパフォーマンス改善の取り組みを紹介します

2021-02-10
ペパボのログ活用基盤『Bigfoot』を使った Zendesk のデータ可視化

2021-02-09
ペパボが気になる人への FAQ

2021-02-05
ステークホルダーマップの活用

技術的難易度の高いチャレンジがあったケースでは、ペパボテックブログに「トラブルシューティング」の記事として昇華します

セキュリティ対策室: これからの課題

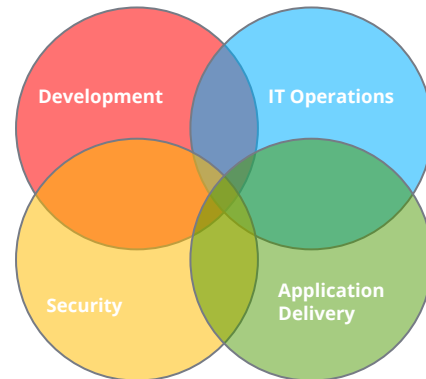
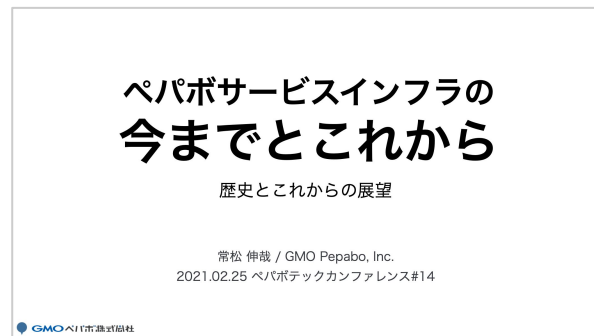
これからもセキュリティの課題は山積み

セキュリティを主軸としていないアプローチでもコミットできる

- コミュニケーションをエンジニアリングで回す
- サービスの安定稼働を目指してSREともオーバーラップしていく

セキュリティを主軸にやりたい人も、もちろん求めている

- 詳細は 森田 @mrtc0 のターンで!





Section 4

セキュリティ対策を継続させるプロダクトの開発

Kohei Morita / @mrctc0

自己紹介

名前と所属

森田 浩平 / @mrtc0

セキュリティ対策室 シニアエンジニア

経歴

~ 2018年 脆弱性診断会社でアルバイト

2018年4月 新卒入社

その他

セキュリティ・キャンプ講師, ステアリングコミッティ

IPA未踏事業 クリエイター

OWASP Fukuoka Chapter Leader



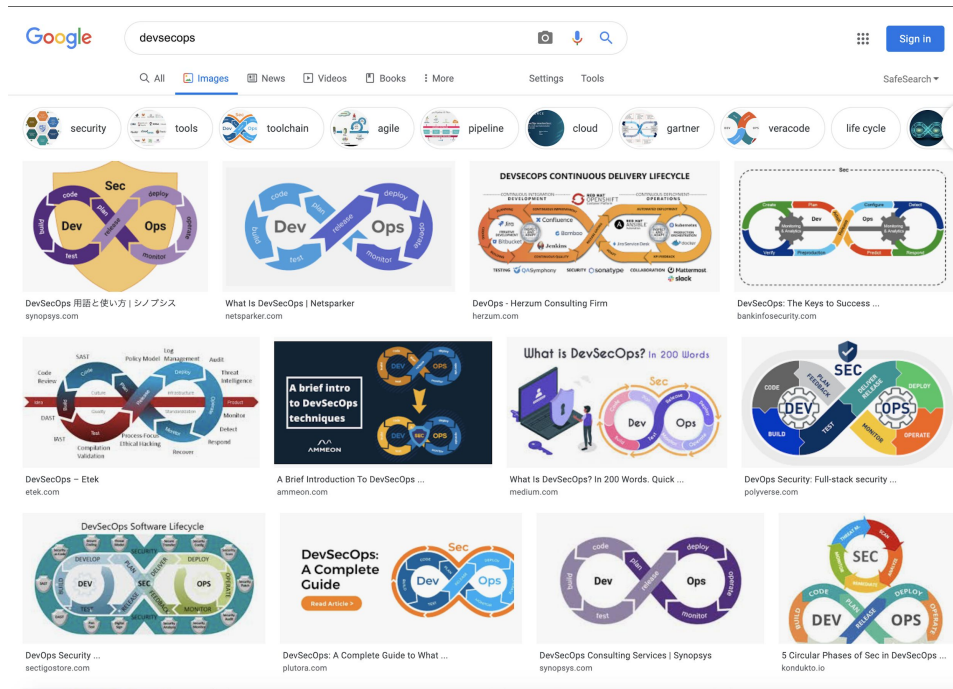
Continuous

攻撃やインシデントがなくなることはないため、対策を **"継続する"** 必要がある

- 攻撃が止むことはない = 防御も止めてはいけない
- 持続的で大きな効果を生み出す = レバレッジの効いた技術を使う / 対策をする
- 継続するために自動化する 🤖

DevSecOps の推進

- 「セキュリティ対策はやって当たり前」というマインドセットへ
- DevOps に Sec が混じっている図
- 自分たちはどこを守れているんだっけ ... ?



DevSecOps の推進

- 自分たちの DevSecOps サイクルを考える
- 不足している領域を穴埋めしていく
- 課題があれば領域を広げていき、層を厚く(多層防御)していく





Section 5

継続させるための エンジニアリング

セキュリティ対策室のエンジニアリング例

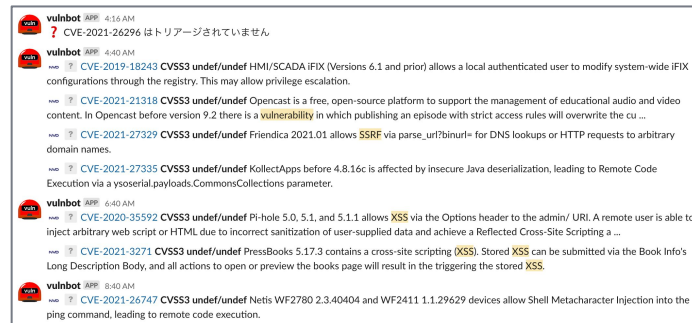
脆弱性情報の収集・トリアージ

組織横断的に脆弱性情報の収集とトリアージを行っている

- CVE・JVNDB のアグリゲーターを実装
 - Slack Bot を通知インタフェースとする
 - Affected 疑いのものは pin 留めして remind

- 対応を要する脆弱性は issue で組織展開
 - 例: CVE-2021-3156 sudo の脆弱性

emoji でもセキュリティチャンネルに情報共有を集う



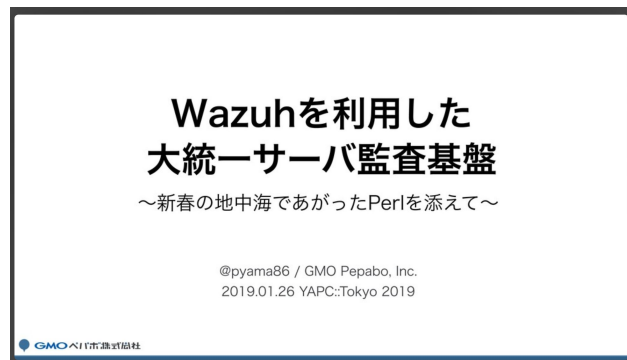
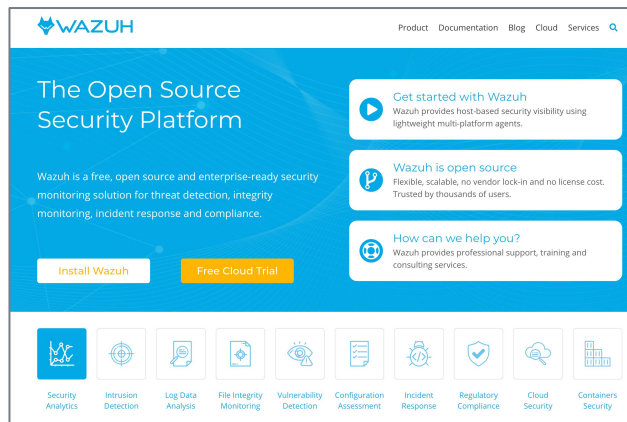
Wazuh の運用

Wazuh (わずー) Agent モデルのセキュリティ基盤

- ログ監視に基づいた侵入検知
- ファイル整合性監視 (FIM)
- ポリシー監査
- 各種インベントリの収集 / 脆弱性検知

Wazuh の構築・導入は 山下 @pyama86 と 常松 @tnmt が進めてくれた。

セキュリティ対策室で運用を固めている。



Wazuh 運用の継続的な改善

- ・組織やチームに適合した運用をするため、不足している箇所を自分たちで作っていく
- ・コミュニティにも貢献していく

Fixed URL anchors #2223

Merged alberpilot merged 1 commit into wazuh:3.11 from hiboma:3.11-fix-url-anchors on 7 Mar 2020

Conversation 2 Commits 1 Checks 0 Files changed 3

hiboma commented on 23 Feb 2020

- Hi. This PR fixes URL anchors for regex.html.
- os-regex-or-regex-syntax -> regex-os-regex-syntax
 - os-match-or-sregex-syntax -> sregex-os-match-syntax

README.md

Wazuh Ruby Client

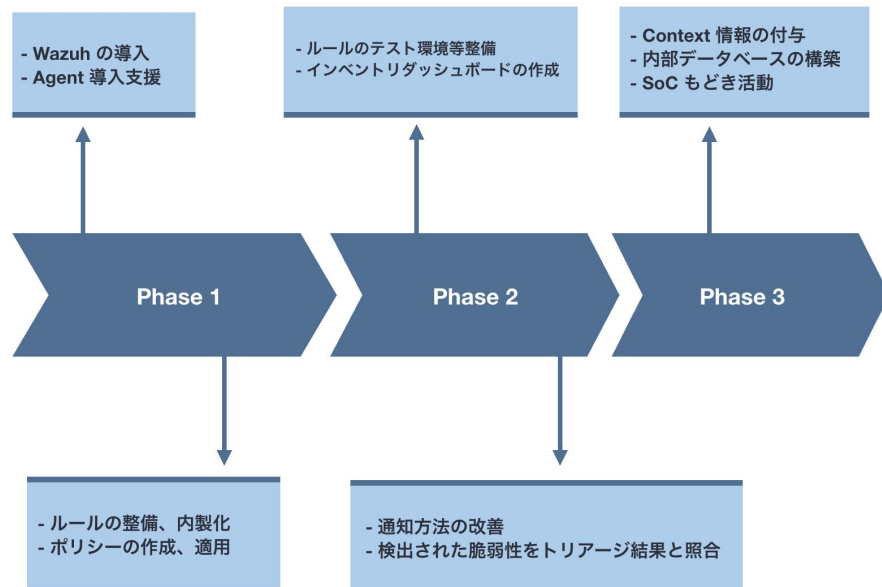
License MIT GitHub Workflow passing yard docs

A Ruby client for the wazuh APIs.

Installation

Add this line to your application's Gemfile:

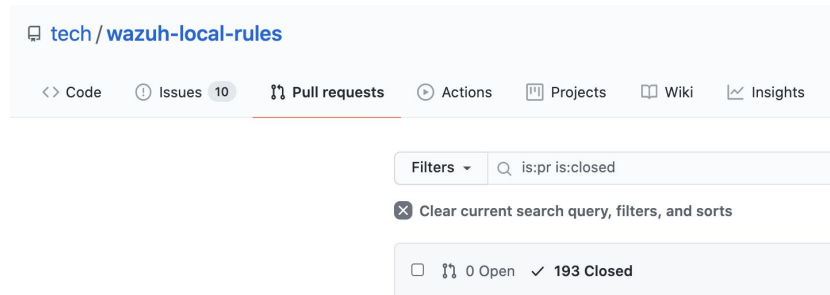
```
gem 'wazuh-ruby-client'
```



Wazuh ルールの継続的な運用

- デフォルトのルールは検知対象が多い
 - 自分たちでカスタマイズしていく必要性
 - Centralized configuration で集中管理
 - セキュリティ対策室がテコ入れして削減

- サービス側もルールの調整を行う
 - ルールを書きやすい環境を整備
 - YAML でルールとテストログを記載



ルールとテストの書き方

実際に記述したルールが想定どおりに動くかテストすることが出来ます。 `local_rules/<サービス名>/<ルールID>-.yaml` にルールとログと、想定出力のレベルを記述するだけでCI時に自動でテストします。

```
rule: |
  <!-- ここに rule を記載する -->
  <rule id=123456789" level="12">
    <... >
  </rule>
tests:
- log: full_log をコピペする
  level: 想定レベル を記載する
- log: "Oct 11 15:03:19 sslcache-10-241-0-69 sshd[24082]: Did not receive identification string from
  level: 0
```


通知のカスタマイズ

- ステージング環境での検証等で生じる一時的なエラーなども検知され、大量に通知が来てしまう
- アラートに対してどのようなアクションを取ればよいかわからない
 - -> 流量や通知内容のコントロールを行うために、通知モデルを実装



⚠ Group Scope Flooding! ルール 1010 の通知が溢れています。通知を drop します

Process segfaulted.

Feb 25 06:27:44 : kernel: [4494341.003069]

👉 単位時間あたりの同一アラート数が溢れたら、そのアラートを止める

Annotation やリンクをつけることで対応を促したり、確認負荷を下げる 👉

General SATA disk failure

Feb 23 14:21:30 : [18045039.399985] ata5.00: failed

command:

Agent

Rule ID
5138 (Level 7)

Location

/var/log/syslog

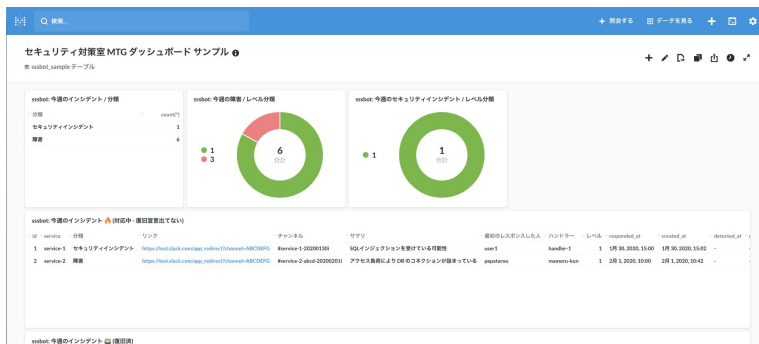
Alert
realtime

Annotation

🌐 ハードウェアトラブル予兆のアラートです

その他のセキュリティ対策室の取り組み

- サービス安定のための技術を検証、導入している。例えば
- 社内 Kubernetes クラスタ管理ツール NKE への Falco や Security Policy の導入
- ロリポップ！ マネージドクラウドのコンテナイメージの検証に Open Policy Agent を導入
- パッケージ等のインベントリを横断的に検索できるアーキテクチャの開発と運用
- Linux カーネル周辺のバグ調査、脆弱性の実現可能性の調査



PANOP

REPOSITORIES

IMAGES

Signed in as mrtc0

Repository List

Full name

Search

REPOSITORY	TYPE	RENOVATE	ACTION
org1/app	Rails	Installed	Edit
org1/app-tools	Ruby	Ignored	Edit
org1/frontend	TypeScript	Required to install	Edit

現状の課題やこれから

アプリケーション

- ・SAST や DAST を利用した脆弱性検出の強化

インフラ

- ・セキュリティポリシーの Enforce

インシデント

- ・プレイブック(対応)の自動化や SOAR への発展

その他

- ・モバイルなど
- ・人的リソース (今は3人しかいない!)

以下、一つでもマッチする方を募集しています！

- 技術力で DevSecOps の推進に貢献してくれる方
- Linux カーネルレイヤの問題に取り組める方
- プロダクトのセキュリティエンジニアリングに関わりたい方
- この発表を聞いて「興味がある！」となった方

「ペパボ 採用」[検索]
<https://recruit.pepabo.com/>

※ セキュリティエンジニアの募集枠は後日出ます
各サービスのソフトウェアエンジニア / SRE としてセキュリティに携わりたい方も大募集！