



JANOGerとクラウドの ワクワクする関係

Yukihiro Kikuchi

2020.01.13



はじめに概要おさらい

クラウドの普及に伴い、JANOGerもクラウド接続に携わったり、サーバプラットフォームとして触れたりする機会が増えてきました。クラウドにも物理/論理のネットワークは必須であり、その内外ではBGPやIPSec VPNといったおなじみのプロトコルが広く利用されています。これらのデザインやオペレーションにJANOGerをはじめとするインフラエンジニアの腕の見せ所、そしてワクワクするポイントが数多くあります。

本プログラムでは、JANOG43,44「インフラエンジニアがクラウドについて語らうBoF」での議論内容もふまえ、インフラエンジニアがどのようにすればクラウドとワクワクする関係を作れるのか、そしてどうすればよりよく活用していいけるのか、運用の勘所やネットワーク技術を中心に、広く深く議論できればと考えております。

スピーカー

- ネットワークエンジニアの 自分がクラウドインフラとどう向き合ってきたか
エクイニクス・ジャパン株式会社
丸野達矢さんから
- いちクラウド提供者側からこれからの「ワクワク」できるネットワークを考えてみる
アマゾン ウェブ サービス ジャパン株式会社
菊池より



JANOG45: JANOGerとクラウドのワクワクする関係

いちクラウド提供者側からこれからの
「ワクワク」できるネットワークを考えてみる

Amazon Web Service Japan K.K.

Yukihiro Kikuchi

2020.01.13

自己紹介

名前：菊池 之裕(きくち ゆきひろ)

所属：アマゾン ウェブ サービス ジャパン株式会社
技術統括本部 レディネス&テックソリューション本部
シニアソリューションアーキテクト ネットワークスペシャリスト

ロール：Network系サービスについてのご支援

経歴：ISP,IXP,VPN運用、開発を経てネットワーク機器、仮想ルータ販売会社のプリセールス、プロダクトSEからAWSへ

好きな AWS サービス:AWS Transit Gateway,
AWS Direct Connect, AWS Marketplace



クラウドはまったく新しい価値観の世界

クラウドでの一般設計原則(Design Principles)

- 必要なキャパシティを勘に頼らない
- 本番規模でのシステムテストを行う
- アーキテクチャ試行の回数を増やすために自動化を取り入れる
- 発展的なアーキテクチャを受け入れる
- データ計測に基づいてアーキテクチャを決定する
- 本番で想定されるトラブルをあらかじめテストし、対策する

今日もって帰っていただきたいこと

信頼性について考えてみる

Design for failure

クラウドに必要な考え方と技術

本質的なことは変わらない

クラウドを使った自動化と高信頼性について

絶えず壊し続ける手法（カオスエンジニアリング）のご紹介

Agenda

- 信頼性について考えてみる
- オンプレミス／クラウド間接続を高信頼にするには
- クラウドに必要な考え方と技術
- クラウドを使った自動化と高信頼性について

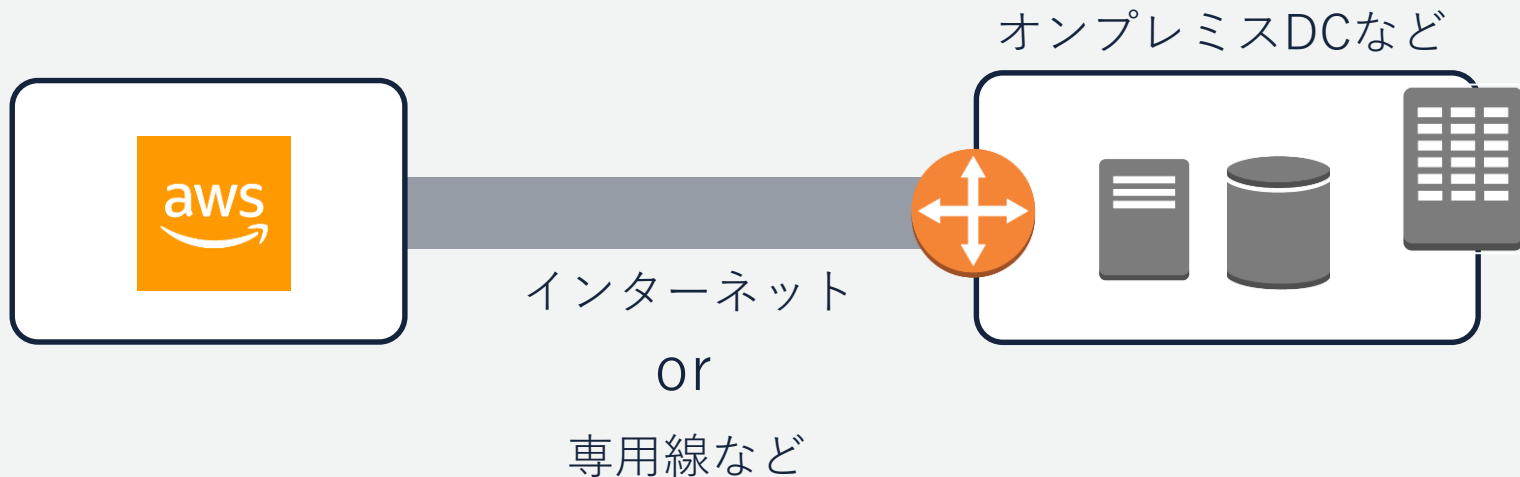
Agenda

- 信頼性について考えてみる
- オンプレミス／クラウド間接続を高信頼にするには
- クラウドに必要な考え方と技術
- クラウドを使った自動化と高信頼性について

信頼性について考えてみる

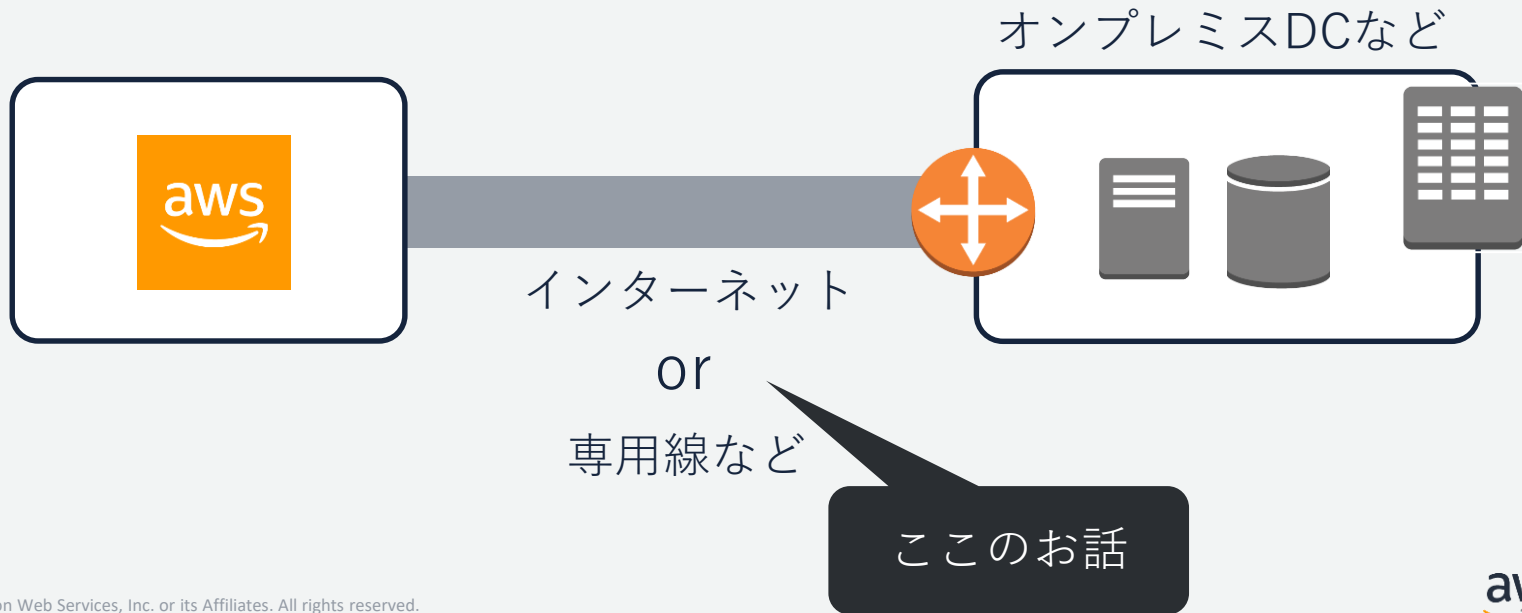
AWSとオンプレミスを接続するためには

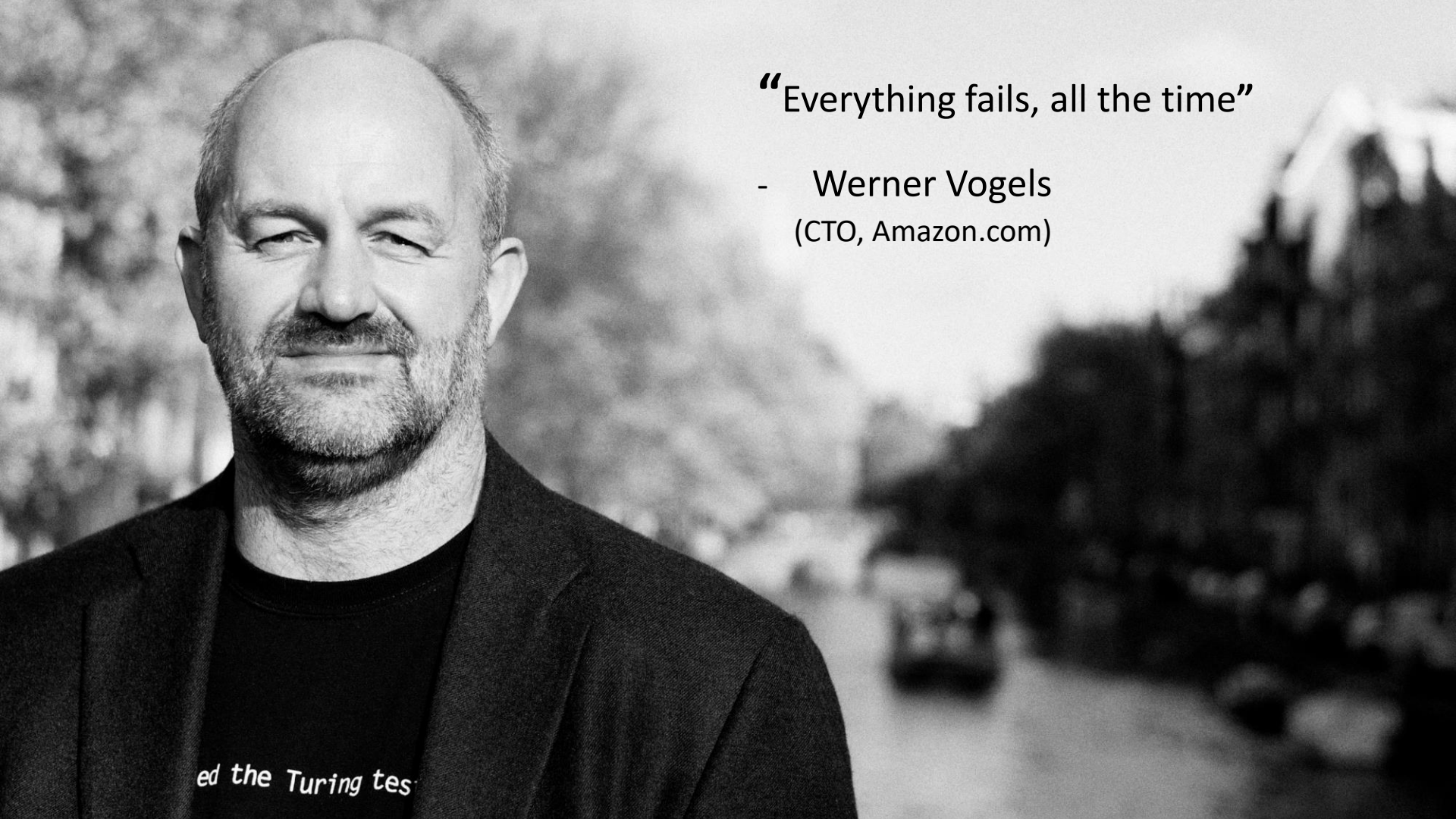
- オンプレミス環境をAWSクラウドに接続するためにはネットワークが必要
- ネットワーク=インターネットまたは専用線など



AWSとオンプレミスを接続するためには

- オンプレミス環境をAWSクラウドに接続するためにはネットワークが必要
- ネットワーク=インターネットまたは専用線など

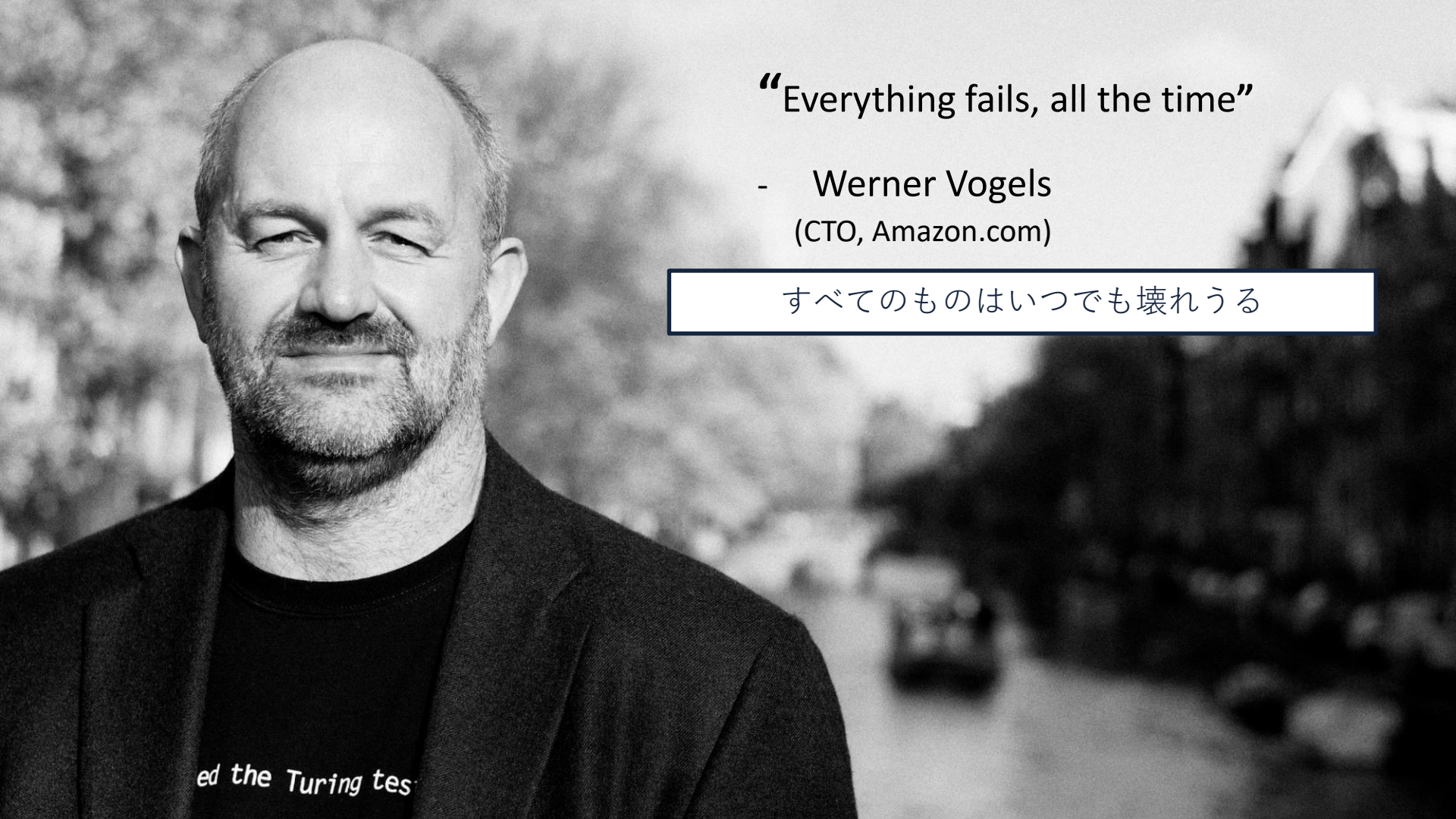




“Everything fails, all the time”

- Werner Vogels
(CTO, Amazon.com)

ed the Turing tes



“Everything fails, all the time”

- Werner Vogels
(CTO, Amazon.com)

すべてのものはいつでも壊れうる

ed the Turing tes

障害事例

米国

工事中の事故で
光ファイバー切断



13

パケットドロップ

障害事例

ブラジル

ゴミ捨て場火災で
光ファイバー切断



お客様より
インパクトの申告なし

リージョン内ネットワークの可用性



ダークファイバー網

低遅延と経路の
多様性に最適化



自社による運用

ファイバーの経路を
定期的に点検



位置のトラッキング

地理空間座標を
用いた配置の分析



キャパシティ拡張

光波長多重化を活用

Agenda

- 信頼性について考えてみる
- オンプレミス／クラウド間接続を高信頼にするには
- クラウドに必要な考え方と技術
- クラウドを使った自動化と高信頼性について

オンプレミス／クラウド間接続を
高信頼にするには

オンプレミス／AWS間接続を高信頼にしたい

解決したい課題

- ・ ミッションクリティカルなワークロードを扱いたい
- ・ オンプレミスとAWS間の接続を高信頼で構築したい

解決パターン

- ・ SPOF(Single Point of Failure) の排除
- ・ 費用対効果の観点で冗長化する箇所を優先付けして設計

オンプレミス／AWS間の冗長化・高信頼設計

その前に・・・冗長の考え方を考えてみる

【よくある要求仕様】

TCPセッションは障害時に即座にバックアップ機に引き継がれる事
➔ データセンター／Availability Zoneを跨いだ時点で不可能



システム全体の正常性の定義を考え直してみる
1パケット落としてもTCPでは再送がかかる
システム全体でリカバリができていれば良しとする

Design For Failure の考え方

”ゼットタイに落ちないシステムはない”

疎結合や、水平に展開できるシステムを目指す
IPアドレスに依存しない。DNSを活用
DNSで複数エントリを書くようにする

詳しくはこちらのホワイトペーパーを
「AWSを用いた耐障害性の高いアプリケーションの構築」

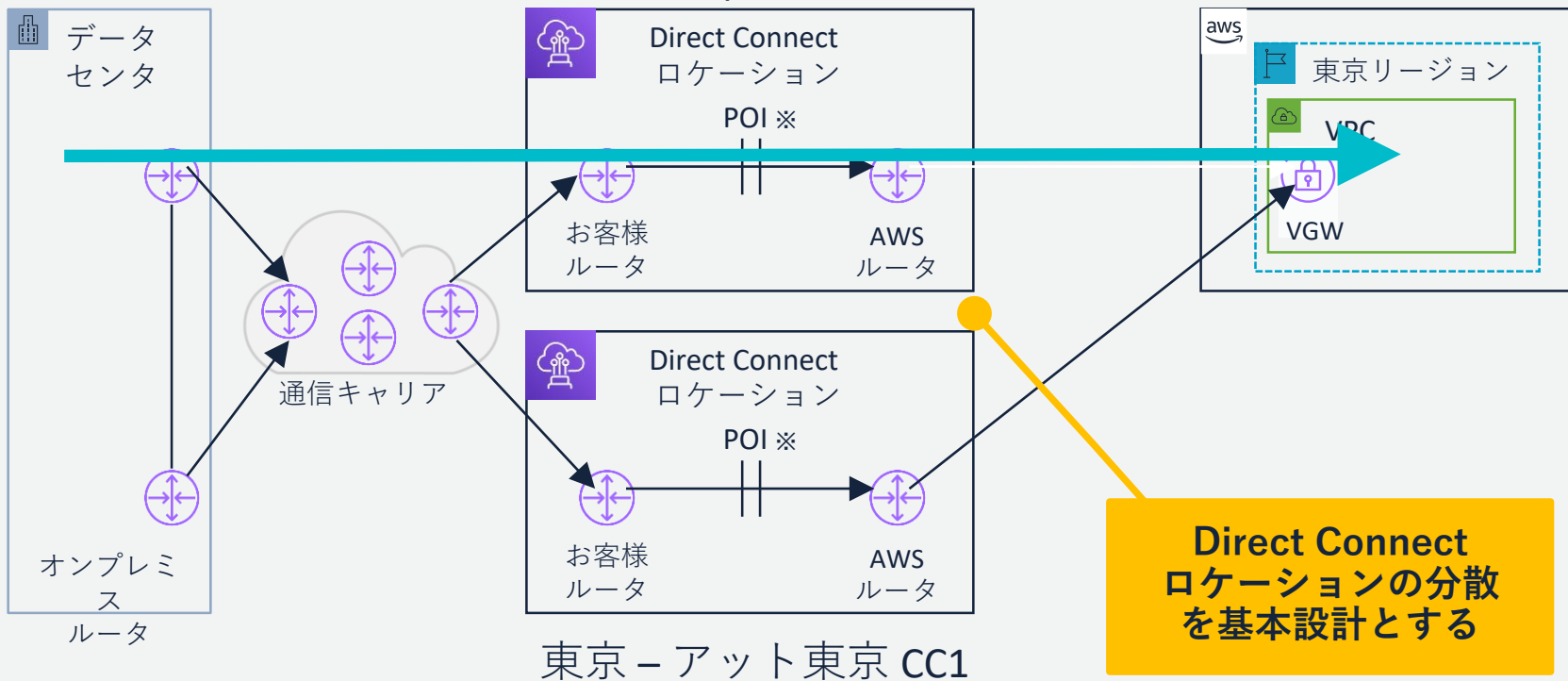
<https://aws.amazon.com/jp/whitepapers/designing-fault-tolerant-applications/>



専用線接続における信頼性

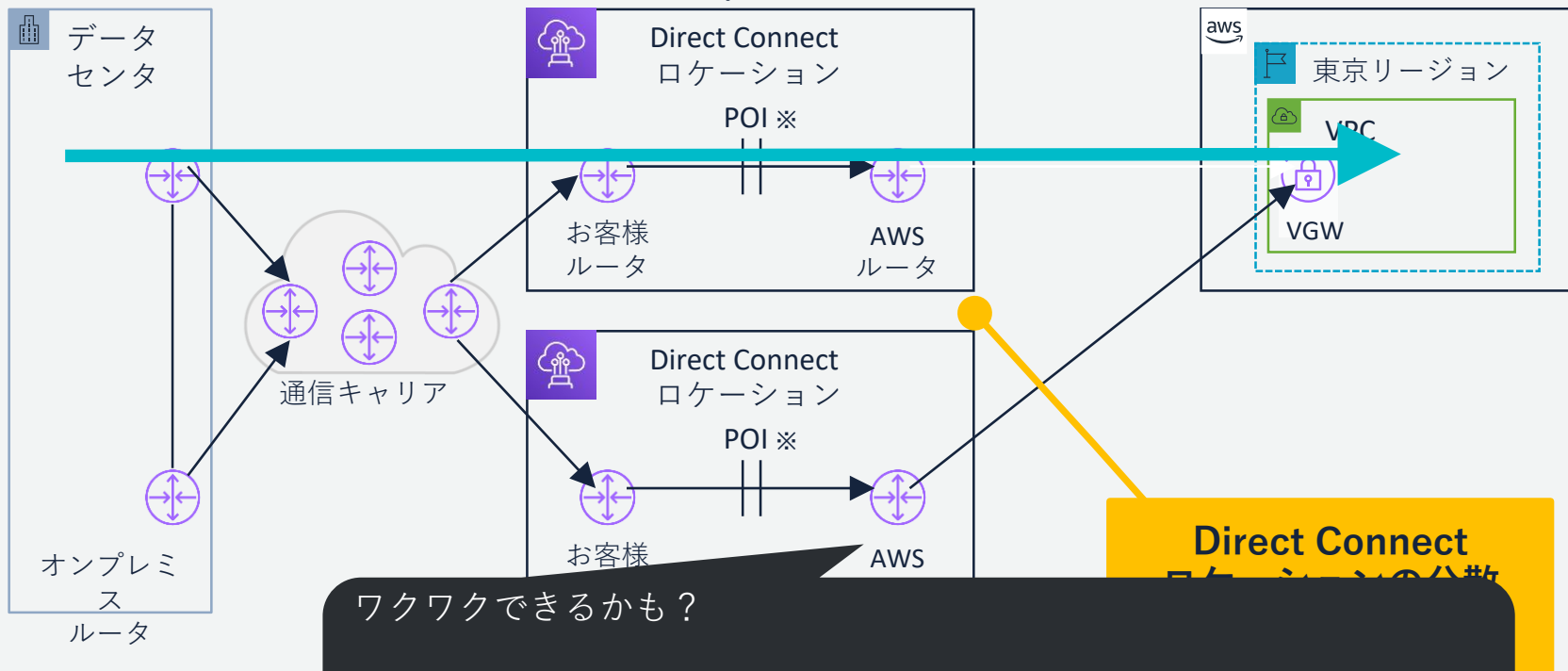
可用性のためのデュアルDirect Connectローケーション

→ トラフィック



可用性のためのデュアルDirect Connectローケーション

→ トラフィック



ワクワクできるかも？

接続点はハードウェアなのでメンテナンスはありうる。クラウド側は壊れても平気なデザインにしているのに張り付き対応を見かけることがある。そこは必要ないのではないか？

信頼性についての定義

- 網としての信頼性 (99.99% ↑)
- アプリケーションとしての信頼性(99.9%)

エンドユーザーが見ている
のはこちら

信頼性についての考え方

- 網としての信頼性 (99.99% ↑)
- アプリケーションとしての信頼性(99.9%)

エンドユーザが見ている
のはこちら

がむしゃらに網の信頼性を上げるよりかは
システム全体の信頼性を見てみては？
アプリケーションの人とお話するとよりワクワクできる関係ができる

Agenda

- 信頼性について考えてみる
- オンプレミス／クラウド間接続を高信頼にするには
- クラウドに必要な考え方と技術
- クラウドを使った自動化と高信頼性について

クラウドに必要な考え方と技術

クラウドはまったく新しい価値観の世界

クラウドでの一般設計原則(Design Principles)

- 必要なキャパシティを勘に頼らない
- 本番規模でのシステムテストを行う
- アーキテクチャ試行の回数を増やすために自動化を取り入れる
- 発展的なアーキテクチャを受け入れる
- データ計測に基づいてアーキテクチャを決定する
- 本番で想定されるトラブルをあらかじめテストし、対策する

ネットワークエンジニアはいらない？でも、、

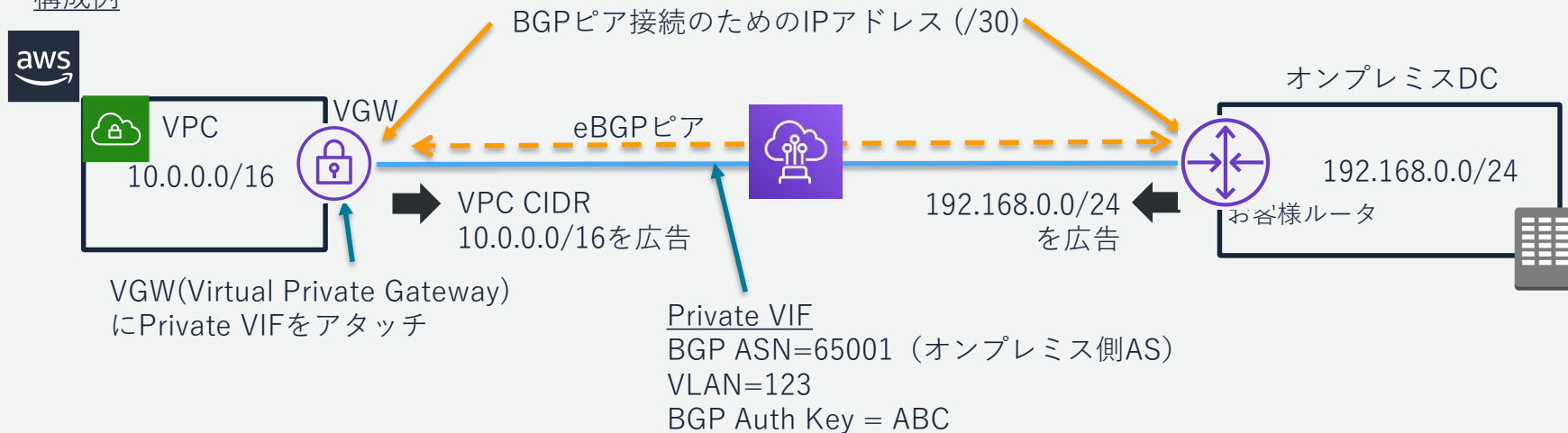
- 本質は変化しない
- 例えば、、

Direct Connect（専用線接続）の場合

- **Private VIF**を使用してVPCへの接続を提供
- お客様ルータでBGP, MD5認証, IEEE802.1q VLANのサポートが必要
- VPCのCIDR(IPv4,IPv6)がAWSから広告される
- Jumbo Frame(MTU=9001)をサポート

https://docs.aws.amazon.com/ja_jp/directconnect/latest/UserGuide/set-jumbo-frames-vif.html

構成例

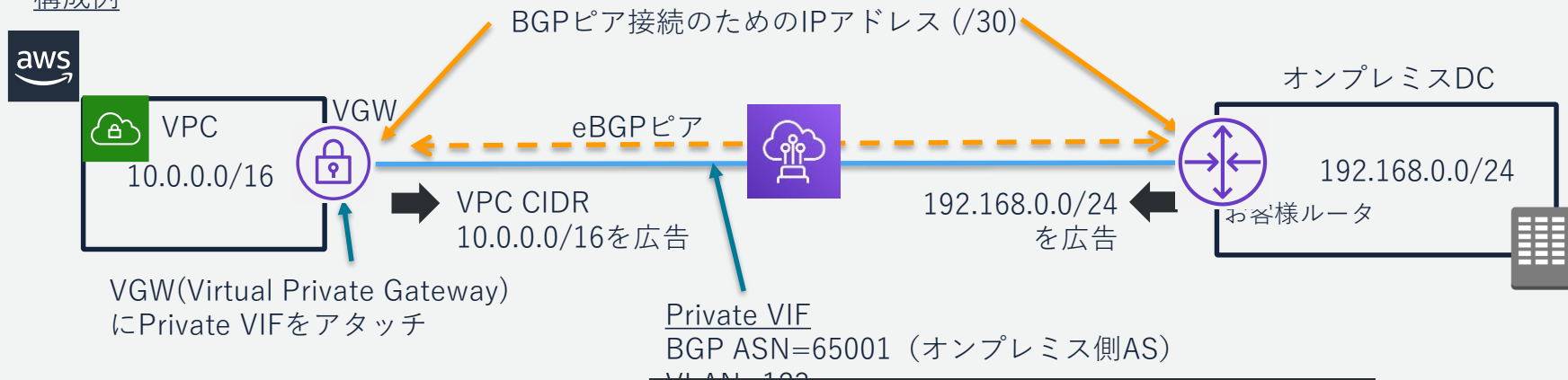


Direct Connect（専用線接続）の場合

- **Private VIF**を使用してVPCへの接続を提供
- お客様ルータでBGP, MD5認証, IEEE802.1q VLANのサポートが必要
- VPCのCIDR(IPv4,IPv6)がAWSから広告される
- Jumbo Frame(MTU=9001)をサポート

https://docs.aws.amazon.com/ja_jp/directconnect/latest/UserGuide/set-jumbo-frames-vif.html

構成例

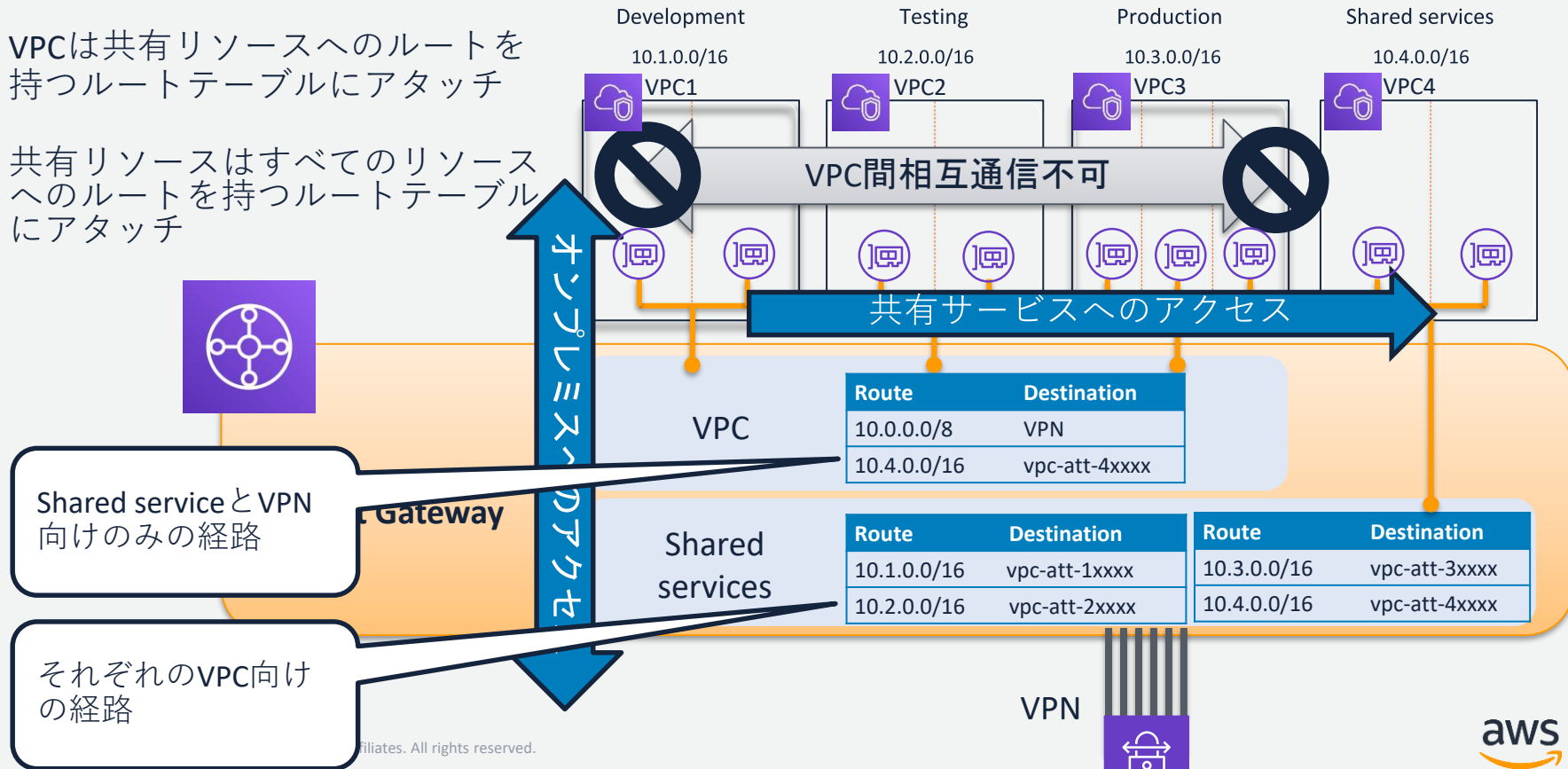


BGPに関する知識が必要

Transit Gatewayの場合

VPCは共有リソースへのルートを持つルートテーブルにアタッチ

共有リソースはすべてのリソースへのルートを持つルートテーブルにアタッチ



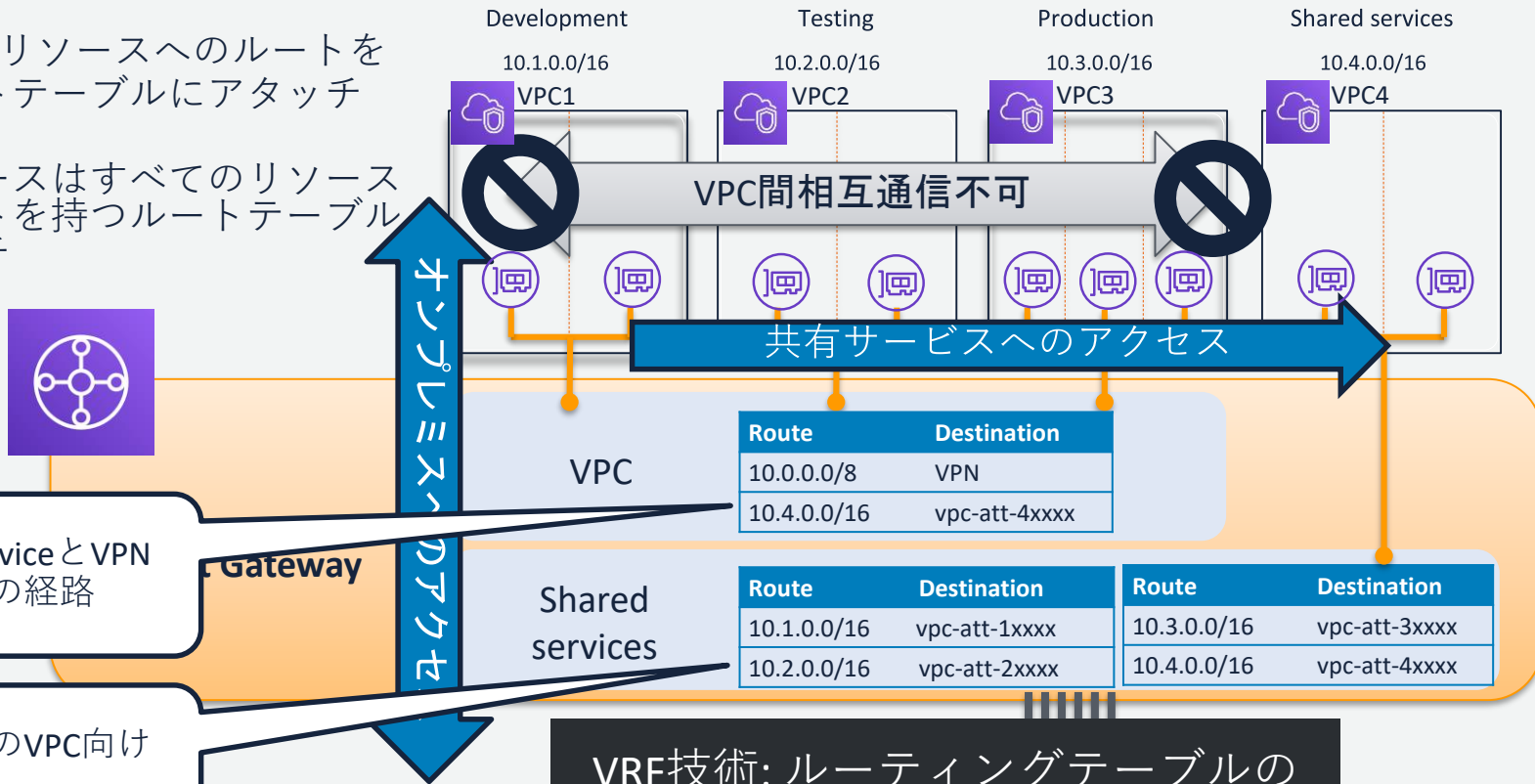
Shared serviceとVPN
向けのみの経路

それぞれのVPC向け
の経路

Transit Gatewayの場合

VPCは共有リソースへのルートを持つルートテーブルにアタッチ

共有リソースはすべてのリソースへのルートを持つルートテーブルにアタッチ



Shared serviceとVPN
向けのみの経路

それぞれのVPC向けの
経路

VRF技術: ルーティングテーブルの
基礎知識が必要

よく見てみると

- 抽象化はされているが、使っている技術は変わらない
 - IPアドレス,ルーティングテーブル、ルーティングプロトコル (BGP)
- クラウドでも引き続きネットワークエンジニアは必要
 - 抽象化されることにより本質の理解が必要
 - 物理機器もAPI (Ansible等)でたたくように

クラウドはまったく新しい価値観の世界（再掲）

クラウドでの一般設計原則(Design Principles)

- 必要なキャパシティを勘に頼らない
- 本番規模でのシステムテストを行う
- アーキテクチャ試行の回数を増やすために自動化を取り入れる
- 発展的なアーキテクチャを受け入れる
- データ計測に基づいてアーキテクチャを決定する
- 本番で想定されるトラブルをあらかじめテストし、対策する

クラウドはまったく新しい価値観の世界（再掲）

クラウドでの一般設計原則(Design Principles)

- 必要なキャパシティを勘に頼らない
- 本番規模でのシステムテストを行う
- アーキテクチャ試行の回数を増やすために自動化を取り入れる

• Network環境を作って壊すのには最適では？
• (わりと) テストの中で失敗を繰り返しやすいのでは？
• 試行してみることがやりやすい

る

ある外資系ハードウェア企業での検証の例

- 実際に機材をラックに積んで配線
- 設定を入れて確認

場合によっては機材がないのでサンノゼに出張して現地で検証

ある外資系ハードウェア企業での検証の例

- 実際に機材をラックに積んで配線
- 設定を入れて確認

場合によっては機材がないのでサンノゼに出張して現地で検証

これが、APIコールやマネージドコンソールですべてが解決できる
何度でも試行できる。
TerraformやAnsibleなどで自動化できる。

Agenda

- 信頼性について考えてみる
- オンプレミス／クラウド間接続を高信頼にするには
- クラウドに必要な考え方と技術
- クラウドを使った自動化と高信頼性について

クラウドを使った自動化と高信頼性について

自動化とカオスエンジニアリング ユースケース:Netflix

Project Nimble/Chaos/Failover

- 8 Minute Failover
- Chaos Kong
- DX Failure in IAD
- Titus Running Wild

“Chaos doesn’t cause problems, it reveals them.”



- 8分間（以内）のマルチリージョンフェイルオーバー
- カオスエンジニアリングによる意図的なリージョン障害 (Chaos Kong)
- Direct Connect障害の高信頼性対応

カオスエンジニアリングとは

<http://principlesofchaos.org/>

本番環境の分散システムが過酷な状況でも耐えられるとの確信を得るために、実験するという取り組み

From:

<https://techlife.cookpad.com/entry/2018/08/02/110000>

本番系で絶えず起こしているのがNetflix

日本企業でも

<https://techlife.cookpad.com/entry/2018/08/02/110000>

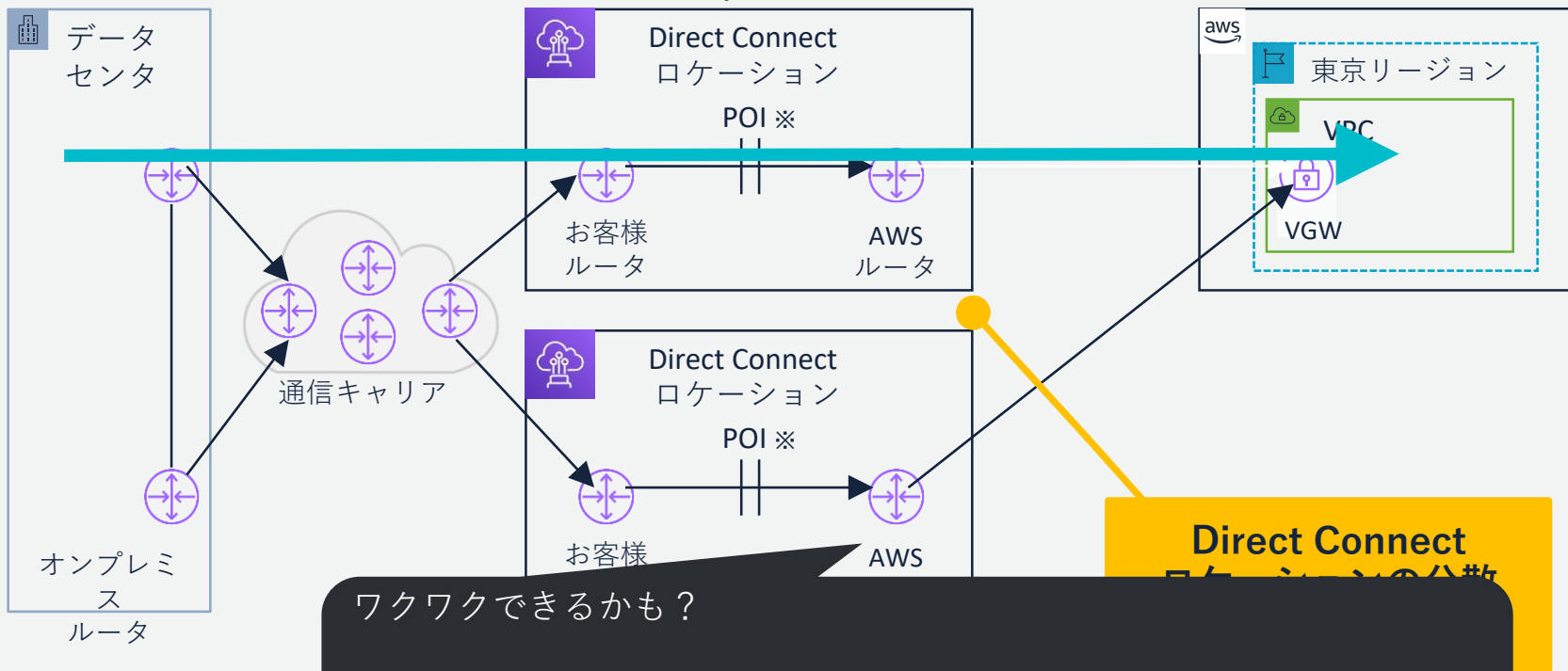
- Chaos Engineering やっていく宣言 by Cookpad



The screenshot shows the header of a blog post on the Cookpad Developers' Blog. The header includes the Cookpad logo (an orange circle with a white chef's hat) and the text 'クックパッド 開発者ブログ' and 'Cookpad Developers' Blog'. Below the header, the date '2018-08-02' is displayed. The main title of the post is 'Chaos Engineering やっていく宣言', which is underlined. The first paragraph of the post reads: '技術部のヨシオリです。Netflix が [Chaos Engineering](#) の論文を公開して 2 年ほど経ちました。クックパッドは最近、Chaos Engineering を導入する事を決めました。この記事ではその背景を紹介したいと思います。' Below this is a sub-section header 'そもそも Chaos Engineering とは'. The text under this header reads: 'Netflix では Failure Injection Testing として、営業時間中に意図的に障害を起す事をやりました。Chaos Monkey というインスタンスとサービスを落とすものから Chaos Gorilla、Kong という availability zone や region 単位で障害を発生させるものなどです。その経験から Chaos Engineering というものが提唱されました。' On the right side of the screenshot, there are two promotional banners. The top one is orange and says 'クックパッドでは エンジニア・デザイナー を募集中で 詳しくはこちら'. The bottom one is green and says 'クックパッド スタッフによる 開発者向け発表資料 詳しくはこちら'. At the bottom right, there is a GitHub logo, the word '検索' (Search), and a search input field with a magnifying glass icon.

(再掲) 可用性のためのデュアルDirect Connectロケーション

→ トラフィック



ワクワクできるかも？

接続点はハードウェアなのでメンテナンスはありうる。クラウド側は壊れても平気なデザインにしているのに張り付き対応を見かけることがある。そこは必要ないのではないか？

問いかけ

- メンテナンス通知時に張り付き対応というフローが変わらない
- メンテナンスも障害の一部、とみなせばメンテナンス張り付きという対応も変わるかも。
- 自動化の取り組みの中でなくせないか？

まとめ

今日もって帰っていただきたいこと

信頼性について考えてみる

Design for failure

クラウドに必要な考え方と技術

基本的なことは変わらない

クラウドを使った自動化と高信頼性について

絶えず壊し続ける手法（カオスエンジニアリング）のご紹介

今日もって帰っていただきたいこと

信頼性について考えてみる

Design for failure

クラウドに必要な考え方と技術

基本的なことは変わらない

クラウドを使った自動化と高信頼性について

絶えず壊し続ける手法（カオスエンジニアリング）のご紹介

考え方は変わっても本質は変わらない

Thank You!

kyukihi@amazon.co.jp

ディスクッション

会場に聞きたいこと

- こんな部分でクラウドをうまく利用してます。などの例があれば知りたい
- 使いたいけど使えない or あえて使ってない。などあればその理由も含めて教えてほしい
- クラウドを使う際CLIベースの運用は必要、ただなかなか教育や文化作りをするの難しい。
うまくやっている例などあれば知りたい

問いかけ

- メンテナンス通知時に張り付き対応というフローが変わらない
- メンテナンスも障害の一部、とみなせばメンテナンス張り付きという対応も変わるかも。
- 自動化の取り組みの中でなくせないか？

補足資料

References

今さら聞けない耐障害性の話 ～クラウドで信頼性を高めるためには～

<https://www.youtube.com/watch?v=9RD7hCZISG0>

<https://pages.awscloud.com/rs/112-TZM-766/images/E3-06.pdf>

AWS re:Invent 2019: Designing for failure: Architecting resilient systems on AWS
(ARC335-R1)

<https://www.youtube.com/watch?v=BJVzwaTiOdk>

AWS の日本語資料の場所「AWS 資料」で検索



The screenshot shows the AWS Japanese website header with the AWS logo, navigation links for '製品', 'ソリューション', '料金', 'ドキュメント', '学習', 'パートナー', 'AWS Marketplace', and 'その他', and a search icon. A '日本語' dropdown menu is also visible. A prominent orange button says 'コンソールにサインイン'. The main heading is 'AWS クラウドサービス活用資料集トップ'. Below it is a paragraph of text in Japanese. At the bottom, there are four buttons: 'AWS Webinar お申込', 'AWS 初心者向け', '業種・ソリューション別資料', and 'サービス別資料'.

aws

日本担当チームへお問い合わせ サポート 日本語 ▼ アカウント ▼ [コンソールにサインイン](#)

製品 ソリューション 料金 ドキュメント 学習 パートナー AWS Marketplace その他 🔍

AWS クラウドサービス活用資料集トップ

アマゾン ウェブ サービス (AWS) は安全なクラウドサービスプラットフォームで、ビジネスのスケールと成長をサポートする処理能力、データベースストレージ、およびその他多種多様な機能を提供します。お客様は必要なサービスを選択し、必要な分だけご利用いただけます。それらを活用するために役立つ日本語資料、動画コンテンツを多数ご提供しております。(本サイトは主に、AWS Webinar で使用した資料およびオンデマンドセミナー情報を掲載しています。)

[AWS Webinar お申込](#) [AWS 初心者向け](#) [業種・ソリューション別資料](#) [サービス別資料](#)

<https://amzn.to/JPArchive>