Privacidade na Internet: Roteamento Aleatorizado com Tor

Semana de Capacitação NIC.br #8

Prof. Dr. Marcos A. Simplicio Jr. – mjunior@larc.usp.br
Escola Politécnica, Universidade de São Paulo

Marcos quem?

- Em suma:
  - Pesquisador em cibersegurança e criptografia desde 2007
  - Professor associado na Universidade de São Paulo (USP) desde 2011
  - Vice coordenador da Comissão Especial em Segurança da Informação e de Sistemas Computacionais (CE-Seg) da Sociedade Brasileira de Computação (SBC)
- Experiência com Tor
  - Usuário desde ~2008 ☺
  - Participação/Coordenação de projetos de pesquisa com tecnologias P2P: FAPESP, CNPq, Ripple, Ericsson, …
  - Autor de cursos online cobrindo Tor: UNIVESP e “Blockchain, Criptomoedas e Tecnologias Descentralizadas”
    - Apoio: Ripple/University Blockchain Research Initiative e Decentralized Foundation – https://www.youtube.com/playlist?list=PLcbbqdJgPgcXlRlPNHF2itTotEwr9kckl

Estrutura da palestra

1. Preliminares: contexto para entender o funcionamento do Tor
   - Fundamentos de segurança e criptografia básica
   - Sistemas descentralizados e DHT
2. Privacidade e Tor
   - Material integrante do curso “Blockchain, Criptomoedas e Tecnologias Descentralizadas”
3. Experimentos com o Tor
   - Navegação privada na web tradicional
   - Navegação privada em domínios .onion
   - Criando um serviço .onion

Cibersegurança e Criptografia: alguns princípios fundamentais

Serviços básicos de segurança

- Definem aspectos fundamentais da segurança de sistemas computacionais.
- São eles:
  - Disponibilidade
  - Confidencialidade
    - Privacidade
  - Integridade
  - Autenticidade
  - Irretratabilidade

Disponibilidade

- Garantia de que usuários legítimos *não sejam impedidos* indevidamente de acessarem as informações e os recursos do sistema.
- Serviço essencialmente extra-criptográfico (físico), e o mais arquitetural/empírico/ heurístico dentre os serviços básicos da segurança.
  - Exemplos de medidas: redundância, controle de acesso (físico), etc.

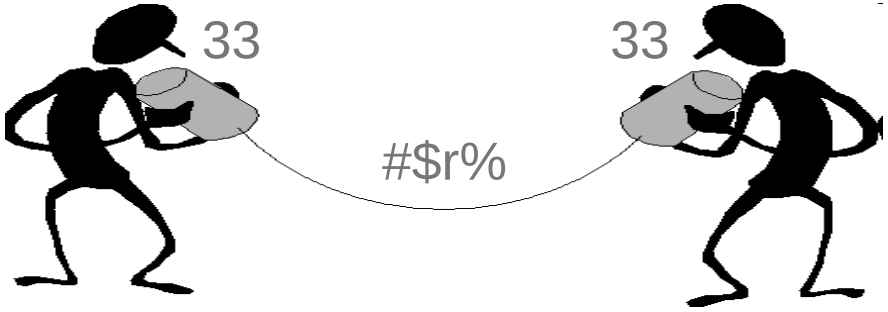
Disponibilidade

- Garantia de que usuários legítimos *não sejam impedidos* indevidamente de acessarem as informações e os recursos do sistema.

No Tor: serviço altamente distribuído (computadores dos usuários mantêm o serviço), sem pontos únicos de falha – elevada disponibilidade por natureza!

Confidencialidade

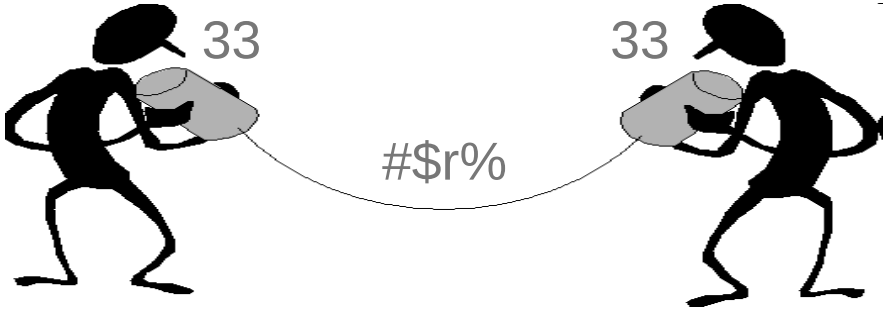
- **Confidencialidade de dados**: garantia de que qualquer *informação* armazenada num sistema de computação ou transmitida via rede seja *revelada somente a usuários devidamente autorizados*.
- Observação: *informação ≠ dado* (representação da informação).
  - Um dado pode estar acessível a qualquer entidade e mesmo assim não revelar a informação que ele contém.



Confidencialidade

- **Confidencialidade de dados**: garantia de que qualquer *informação* armazenada num sistema de computação ou transmitida via rede seja *revelada somente a usuários devidamente autorizados*.

No Tor: mensagens que trafegam na rede são protegidas por confidencialidade



Confidencialidade

- **Privacidade**: Garantia de que os indivíduos *controlem* ou influenciem quais *informações sobre eles* podem ser coletadas e armazenadas e *por quem* e *para quem* tais informações podem ser reveladas
  - Tem relação direta com **confidencialidade de dados** (proteção da informação), mas também envolve políticas de **uso de dados** e **confidencialidade de identidades**.

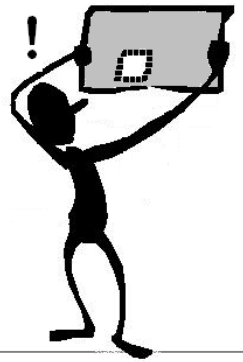
Confidencialidade

- **Privacidade**: Garantia de que os indivíduos *controlem* ou influenciem quais *informações sobre eles* podem ser coletadas e armazenadas e *por quem* e *para quem* tais informações podem ser reveladas

No Tor: endereços IP dos usuários ficam escondidos enquanto eles se comunicam

Integridade

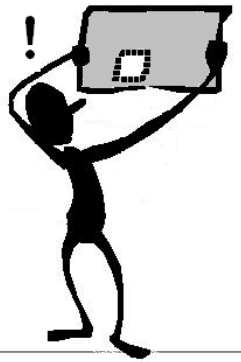
- Possibilidade de *verificar a consistência* da informação contida nos dados, *impedindo que seja alterada* indevidamente de *maneira imperceptível*.
- Detalhe: o serviço de integridade *não* garante que os dados não sejam alterados. A garantia efetiva é que, se os dados forem alterados sem autorização, a alteração será sempre *detectada*.



Integridade

- Possibilidade de *verificar a consistência* da informação contida nos dados, *impedindo que seja alterada* indevidamente de *maneira imperceptível*.

No Tor: mensagens enviadas pela rede têm sua integridade verificada antes de serem entregues a seu destino



Autenticidade

- Garantia de que a *origem* ou o *originador* de uma mensagem seja *corretamente identificado* pelo seu destinatário.
- A *verificação de autenticidade* é necessária após todo processo de identificação, seja de um usuário para um sistema, de um sistema para o usuário ou de um sistema para outro sistema.

Autenticidade

- Garantia de que a *origem* ou o *originador* de uma mensagem seja *corretamente identificado* pelo seu destinatário.

No Tor: dados trocados na web tradicional (HTTPS), identidade de servidores de diretório, nome de domínio de serviços escondidos, ...

Irretratabilidade

- O *originador* e o *destinatário* das informações *não podem negar a sua transmissão, recepção ou posse*.
  - Obs.: ausência de serviço pode ser requisito de segurança (negação plausível)
- Relacionado a *assinaturas digitais*.
  - Conceito similar a assinaturas manuais, mas com garantias matemáticas...

Irretratabilidade

- O *originador* e o *destinatário* das informações *não podem negar a sua transmissão, recepção ou posse*.
  - Obs.: ausência de serviço pode ser requisito de segurança (negação plausível)
- Relacionado a *assinaturas digitais*.

No Tor: assinatura na identidade de serviços – nomes de domínio HTTPS, domínios onion, servidores de diretório confiáveis

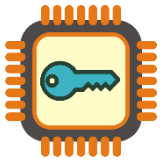
Para que serve criptografia?

- Serviços básicos da segurança:
  - Confidencialidade
  - Integridade
  - Autenticidade
  - Irretratabilidade
- **Não** é possível implementar disponibilidade
  - Mas é exatamente nesse quesito que **sistemas descentralizados**, como Tor, são excelentes!

➔ P2P+Criptografia: um par perfeito ➔

Algoritmos Criptográficos

- Existem dois tipos básicos de algoritmos criptográficos
  - **Simétricos:** uma mesma informação secreta (chave) é conhecida apenas por remetente e destinatário, mas não por atacantes
    - Esta categoria também inclui algoritmos auxiliares, que não usam chaves

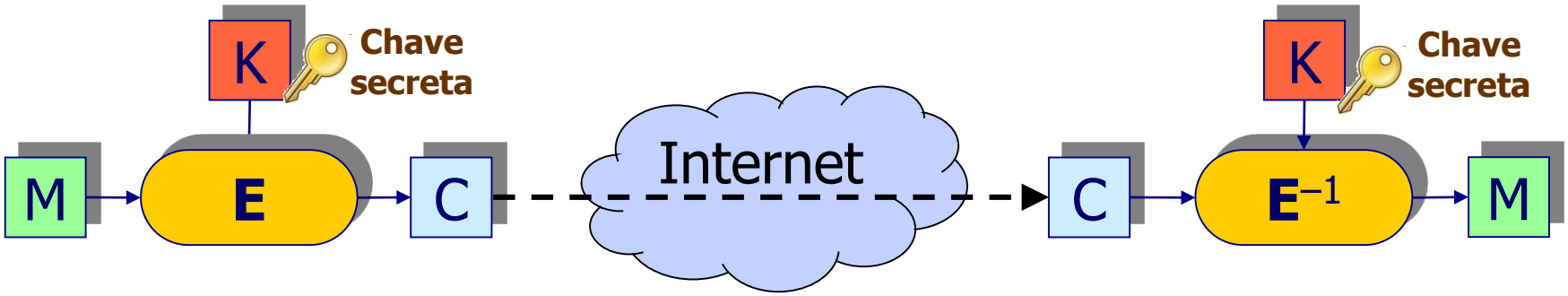


  - **Assimétricos:** usam duas chaves distintas, porém relacionadas matematicamente. Uma chave é tornada pública (conhecida inclusive por atacantes), e a outra é conhecida apenas pelo seu dono.

- Se usada corretamente**, criptografia costuma ser a parte mais forte** de sistemas computacionais

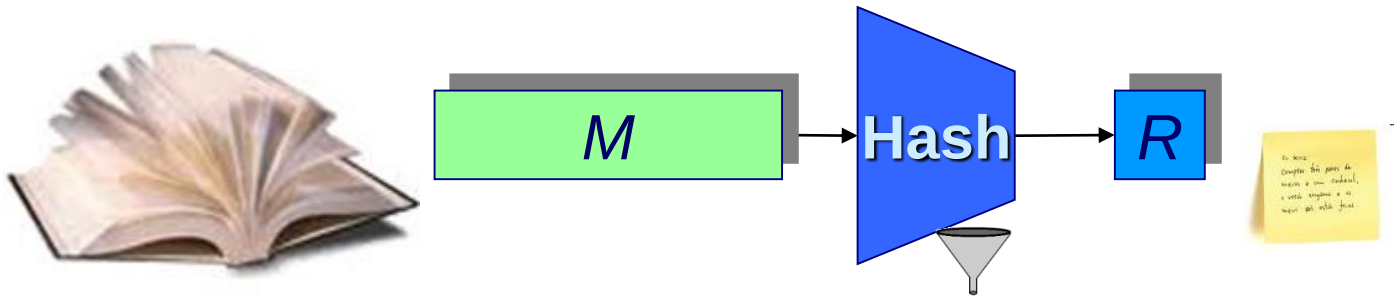
Cifra Simétrica: confidencialidade

- Transformação matemática inversível cujo cálculo depende, no sentido direto (**cifração**) e no sentido inverso (**decifração**), de uma *mesma* informação secreta: a chave K.
  - Se K for descoberta, a confidencialidade é perdida



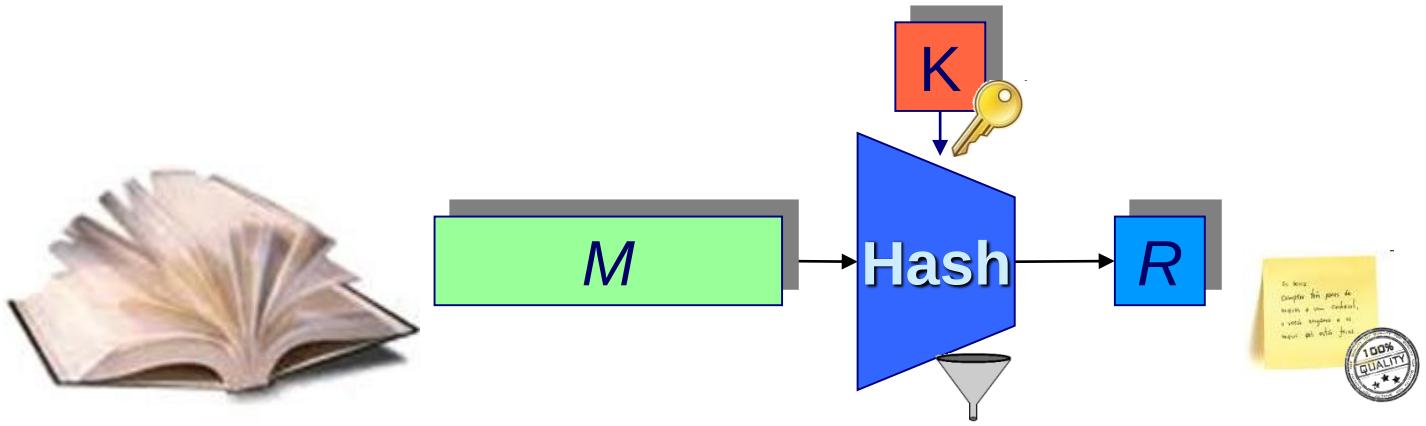
Funções de Hash: Integridade

- Geram um "**resumo criptográfico**" da entrada
  - **Alterações** nos dados de entrada são **detectadas** porque elas **alteram o resumo**
  - O hash tem **tamanho fixo**, e seu valor só depende da mensagem (**não envolve chave secreta**)



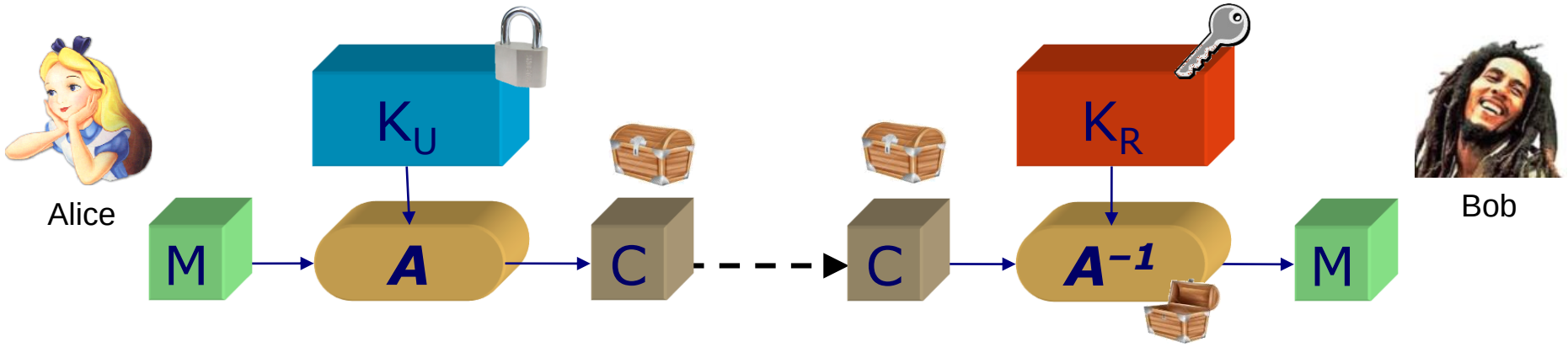
Integridade+ autenticidade

- Hash: "**resumo criptográfico**" da entrada
  - **Alterações** nos dados de entrada são **detectadas** porque elas **alteram o resumo**
  - O hash tem **tamanho fixo**, ~~e seu valor só depende da mensagem (**não envolve chave secreta**)~~
    - **Uso com chave secreta fornece autenticidade**: apenas detentor da chave consegue gerar/verificar hash correto!



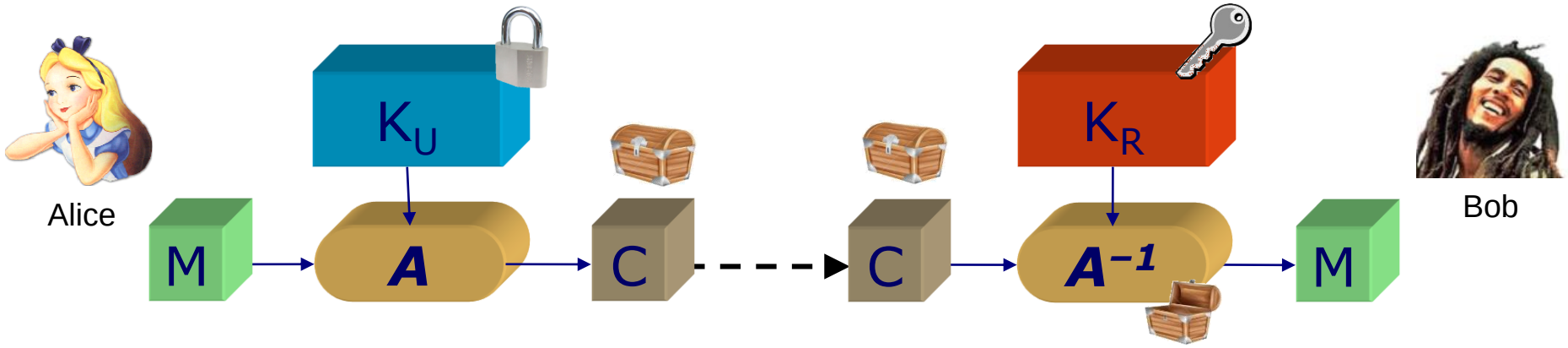
Criptografia Assimétrica

- Cifração: confidencialidade

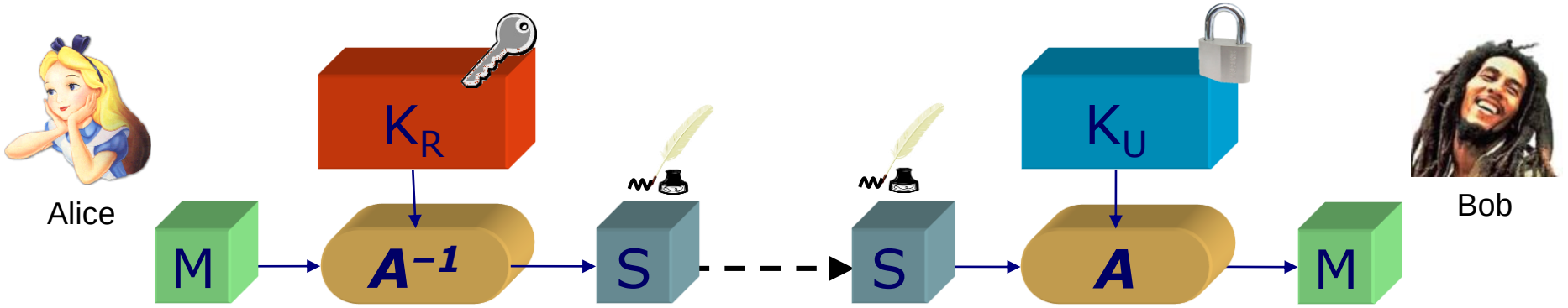


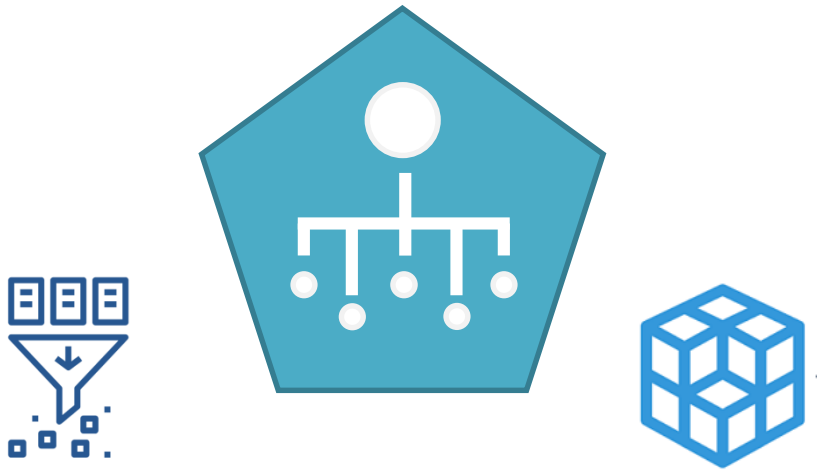
Criptografia Assimétrica

- Cifração: confidencialidade



- Assinatura digital: integridade, autenticidade e irretratabilidade

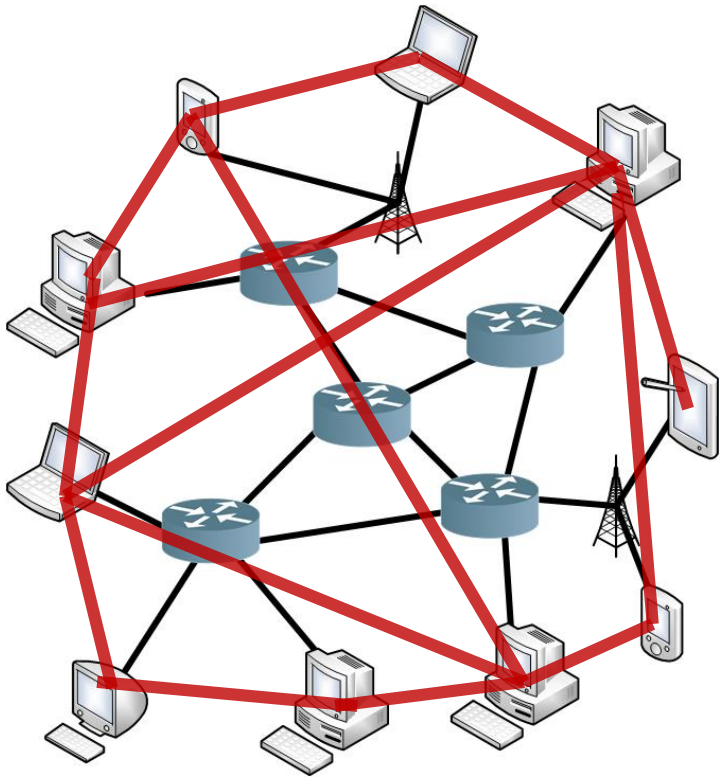




Sistemas descentralizados: alguns princípios fundamentais

Redes P2P: overlay

- Redes P2P são redes "sobrepostas" ("*overlay*")
  - Uma rede formada por **conexões lógicas** entre nós sobre um conjunto conexões físicas existentes
  - **Proximidade física** não é necessariamente levada em consideração
  - Manutenção do overlay pode ser problemática devido à **entrada e saída dinâmica** de nós



Conexão lógica

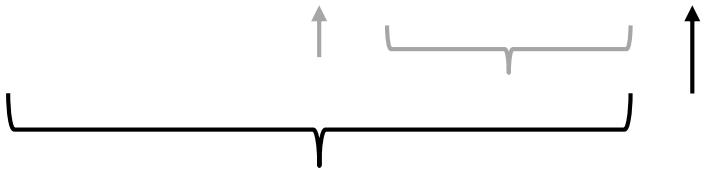
Busca em rede distribuída: DHT

- Recap: Busca em memória local
  - Busca não ordenada: O(n)

33	45	43	17	91	15	71	86	6	62	21

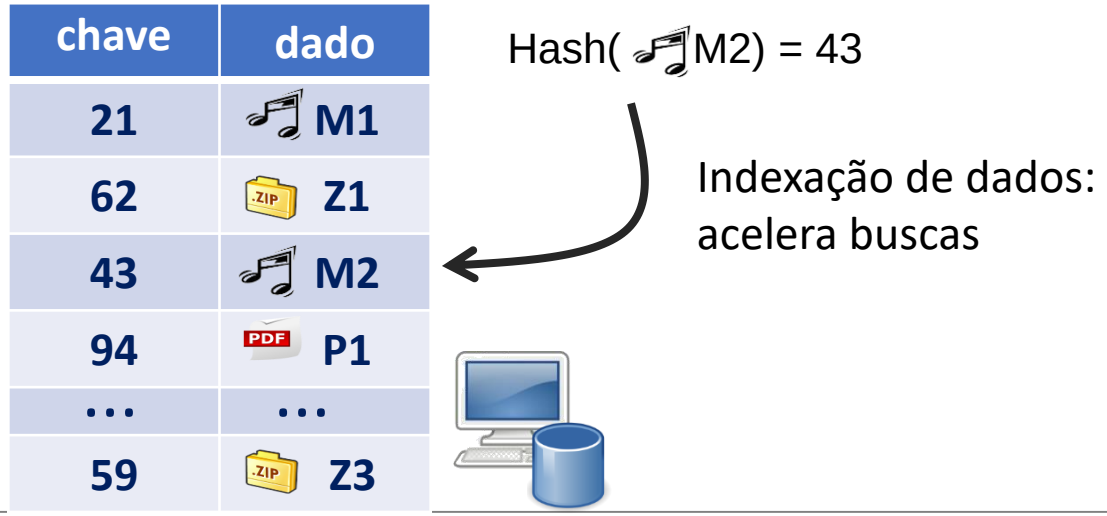
  - Busca binária: O (lg n)

6	15	17	21	33	43	45	62	71	86	91



Busca em rede distribuída: DHT

- Recap: tabela hash
  - Mapeia “chaves” em “valores”: buscas em O(1)
- Chamadas:
  - chave = hash(dado)
  - put(chave, valor): insere valor na tabela
  - get(chave): recupera valor da tabela



Busca em rede distribuída: DHT

- Tabela hash distribuída (DHT): similar a tabela hash, mas espalhada na rede
  - chave = hash(dado)
  - buscar(chave) : IP_nó ➔ O(lg n)
  - rotear(IP_nó, PUT, chave, dado)
  - rotear(IP_nó, GET, chave) : dado

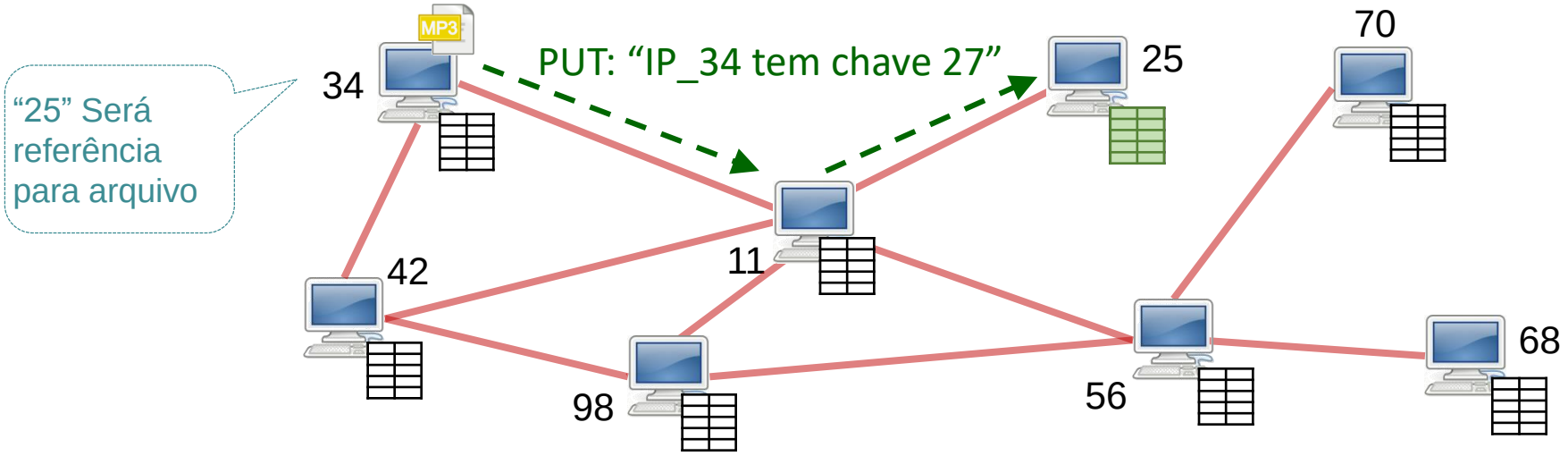
chave	dado
21	M1
62	Z1
43	M2
94	P1
...	...
59	Z3

Hash(M2) = 43

Exemplos de algoritmos de "Rotear": Kademlia, CAN, Chord, Kelips, Pastry & Tapestry

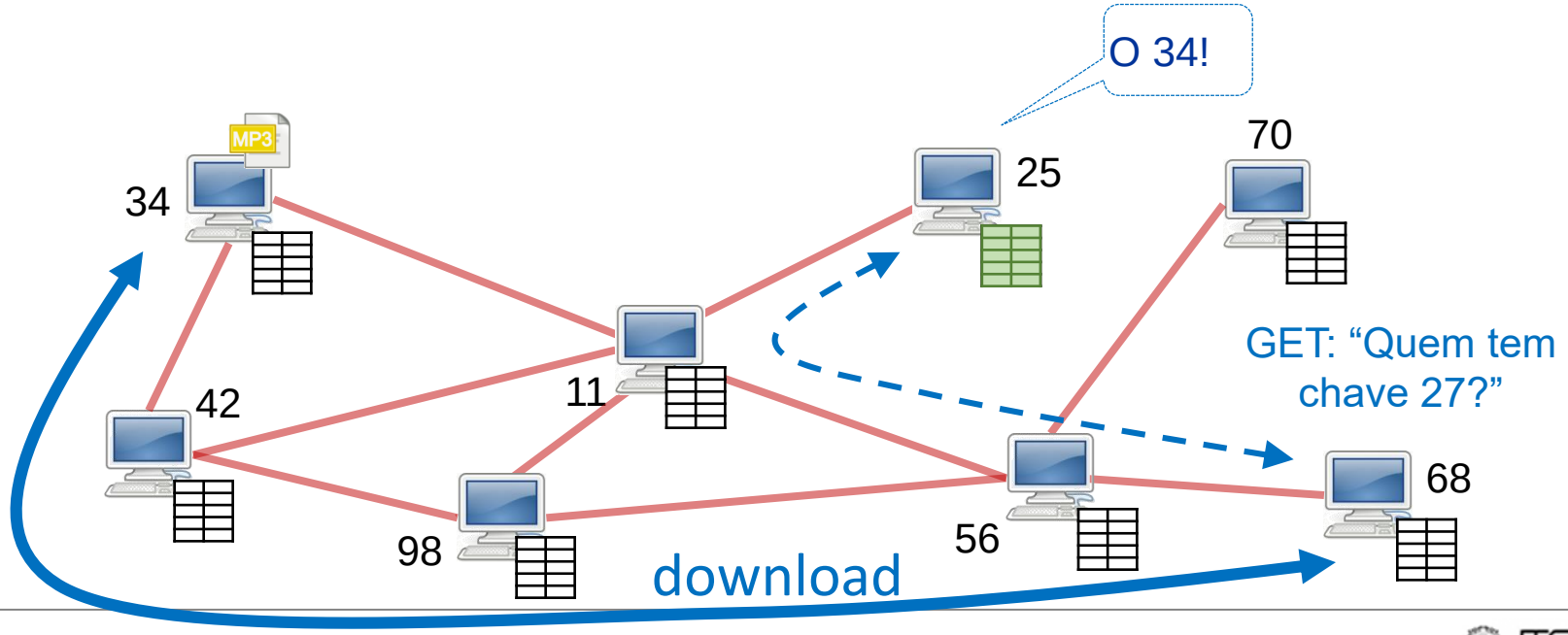
Busca em rede distribuída: DHT

- Tabela hash distribuídas (DHT): similar a tabela hash, mas espalhada na rede
  - chave = hash( ) = 27
  - buscar(chave) : IP_25
  - rotear(IP_25, PUT, chave, IP_34)

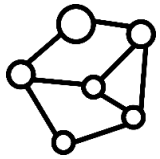


Busca em rede distribuída: DHT

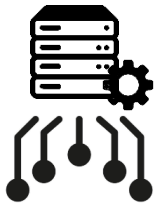
- Tabela hash distribuídas (DHT): similar a tabela hash, mas espalhada na rede
  - chave = hash( ) = 27
  - buscar(chave) : IP_25
  - rotear(IP_25, GET, chave) : IP_34



DHT: quem gerencia?



- Abordagem 1: DHT baseada em peers
  - Nós executando aplicação também mantêm DHT
  - Comum em sistemas descentralizados



- Abordagem 2: DHT infraestruturada
  - Conjunto de nós gerenciados mantêm DHT
  - Em alguns casos, pode dar suporte a mais de um tipo de aplicação

- **No Tor**: abordagem 1, com nós que fiquem longo período na rede (> 24h)
  - Solução similar a Chord: https://doi.org/10.1049/iet-ifs.2015.0121

Privacidade na Internet: Roteamento Aleatorizado com Tor

Funcionamento do Tor

(gravação original do curso Blockchain, Criptomoedas e Tecnologias Descentralizadas)

Blockchain, Criptomoedas & Tecnologias Descentralizadas

Tecnologias descentralizadas:

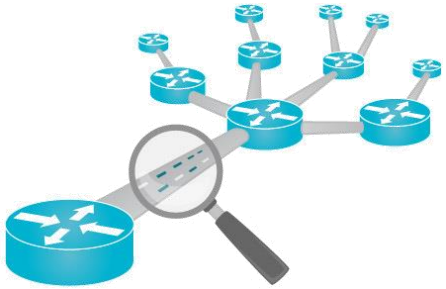
Tor e Privacidade

Prof. Dr. Marcos A. Simplicio Jr. – mjunior@larc.usp.br
Escola Politécnica, Universidade de São Paulo

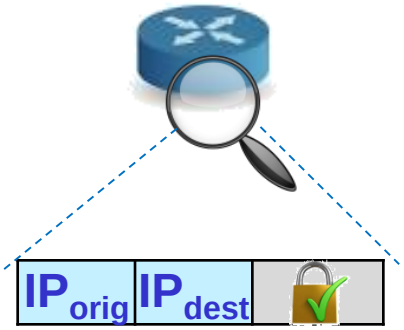
Objetivo:

Pergunta: existe anonimato na Internet?

Roteamento na Internet

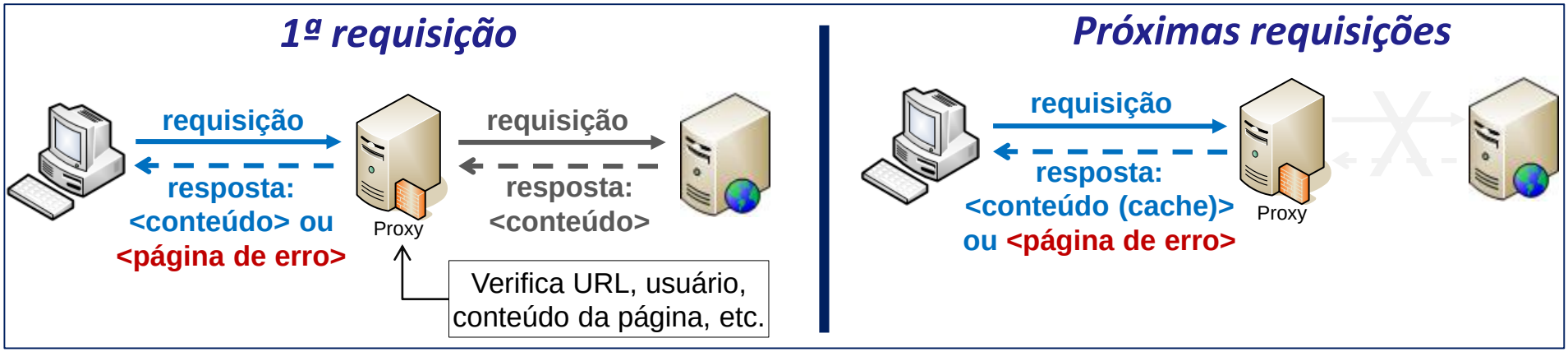


- Nenhuma medida de segurança
  - Tráfego não é cifrado ou autenticado
  - Além da origem e do destino, todos os roteadores intermediários têm acesso ao tráfego
- Solução simples: **s**egurança nas camadas superiores
  - Ex.: HTTP**S**, TL**S**, **S**SH, **S**FTP/**S**CP, etc.
  - Dão confidencialidade, mas não **privacidade**: roteadores intermediários (talvez desonestos) ainda sabem **quem são os nós comunicantes**
- Roteamento com privacidade?
  - "Como disfarçar a origem e o destino dos dados?"

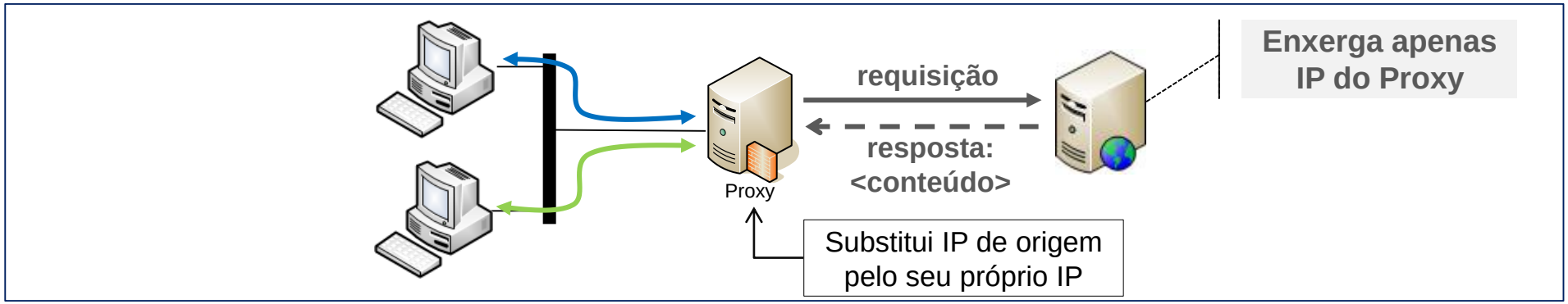


Proxies e privacidade

- Empresas costumam usar **proxies**
  - Objetivos: monitorar conteúdo; otimizar uso de banda



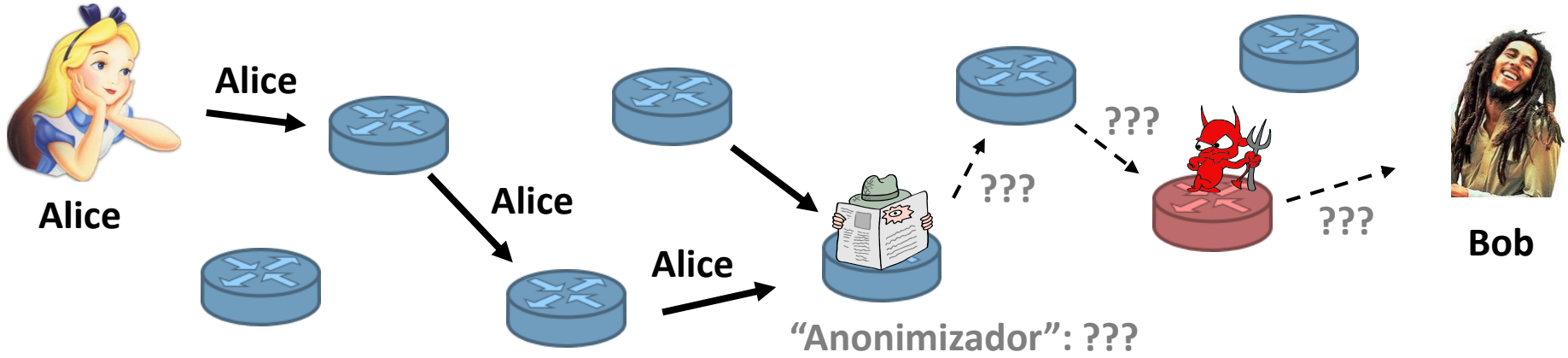
  - Mas também melhoram privacidade da navegação na Internet devido a NAT (Network Address Translation)



Proxies e privacidade

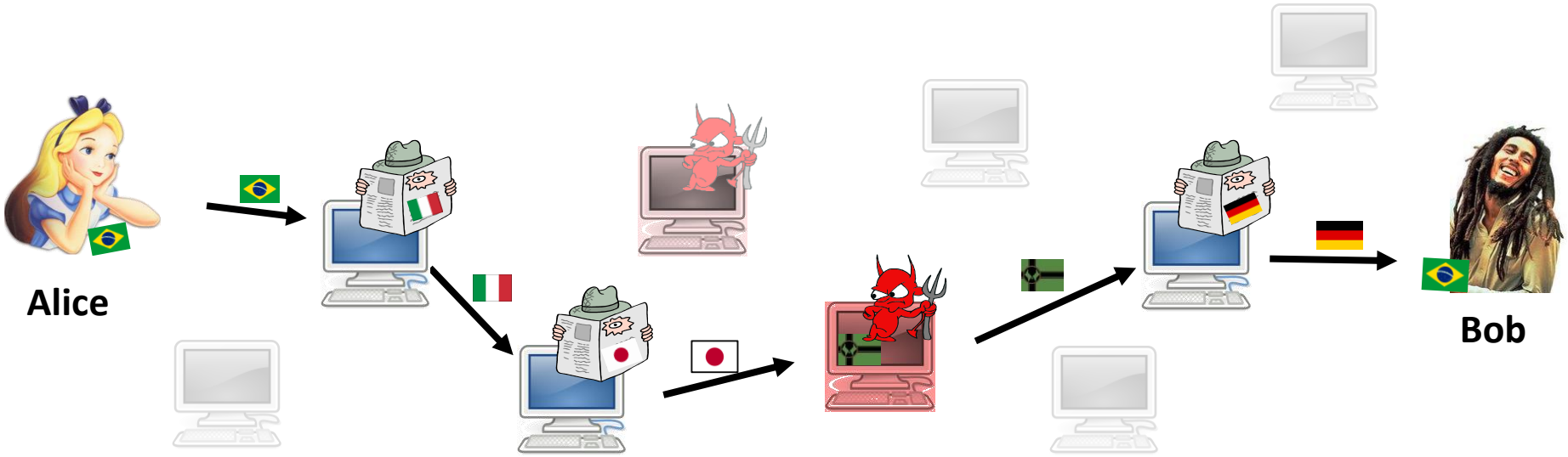
- Proxies web: fazem um serviço semelhante
  1. Proxy acessa página pedida, com IP do próprio Proxy
  2. Página acessada responde normalmente (preferências, como linguagem, são aquelas configuradas pelo Proxy)
  3. Proxy entrega resultado para usuário dentro de seu próprio site (conteúdo HTML do site gerado dinamicamente)
- Conexões normalmente são cifradas (TLS/VPN)
  - Podem também incluir recursos extras, como esteganografia
  - Exemplo: https://hide.me/en/proxy
- Vamos generalizar (e melhorar!) a ideia
  - Afinal, o Proxy pode registrar requisições, ou ser tirado do ar

Generalização: roteamento aleatorizado



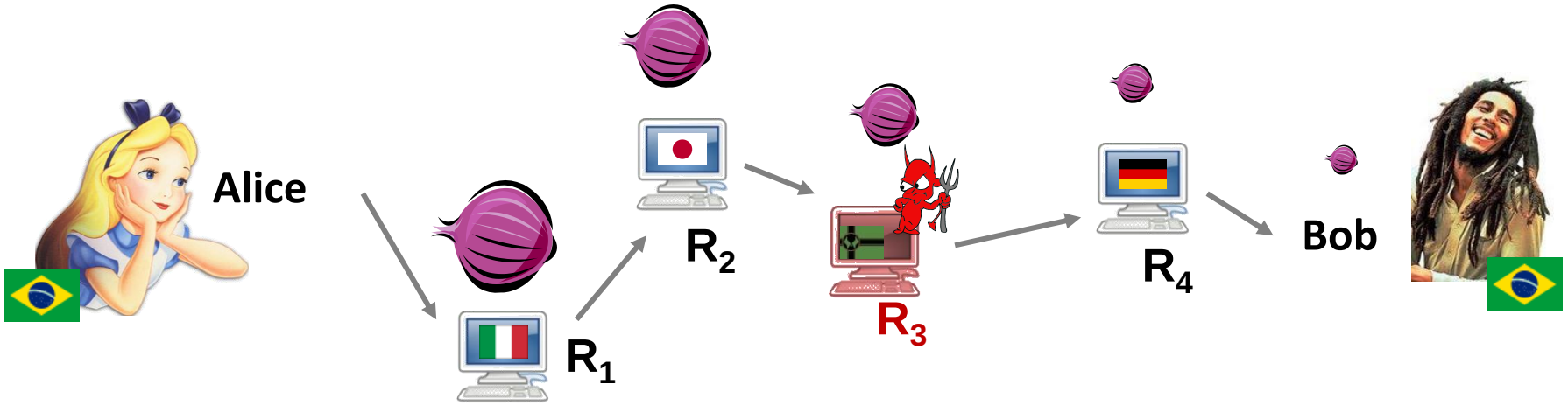
- Disfarça origem das mensagens fazendo roteamento aleatório
  - Técnicas populares: Crowds, Freenet, Onion routing
  - Roteadores não têm certeza se a origem aparente de uma mensagem é de fato seu originador ou outro roteador

Onion Routing

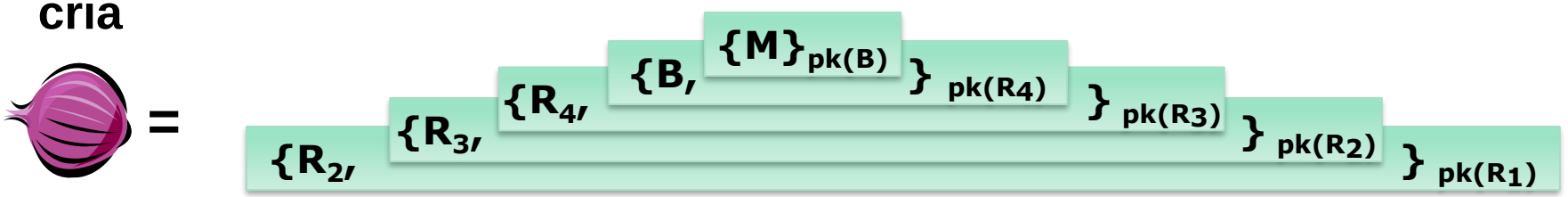


- Origem escolhe uma sequência aleatória de roteadores
  - Alguns roteadores são honestos, outros são controlados por atacantes
  - Origem decide a comprimento do caminho

Onion Routing



cria



Info de roteamento de cada link é cifrada com a chave pública (pk) do roteador

Cada roteador descobre apenas a identidade do próximo roteador

Apenas destinatário acessa mensagem M

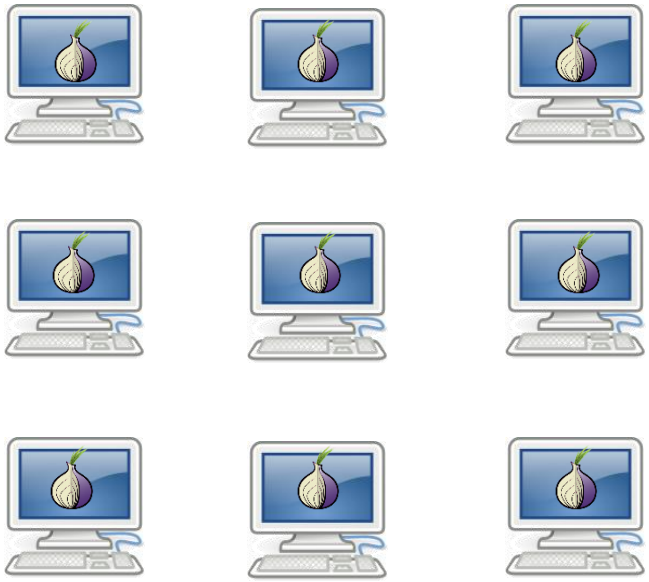
Tor

- Segunda geração do onion routing
  - http://tor.eff.org
  - Desenvolvido by Roger Dingledine, Nick Mathewson and Paul Syverson
  - Projetado especificamente para comunicações na Internet que requerem baixa latência
- Ativo desde Outubro de 2003
  - Diversos nós espalhados pelo mundo todo
  - Milhares de usuários
  - Clientes de "fácil uso" (plugins, Tor Browser)
  - Navegação anônima e gratuita

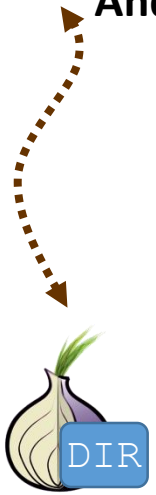
Tor: resumo

Não necessariamente cifrado
Cifrado pelo protocolo Tor

André

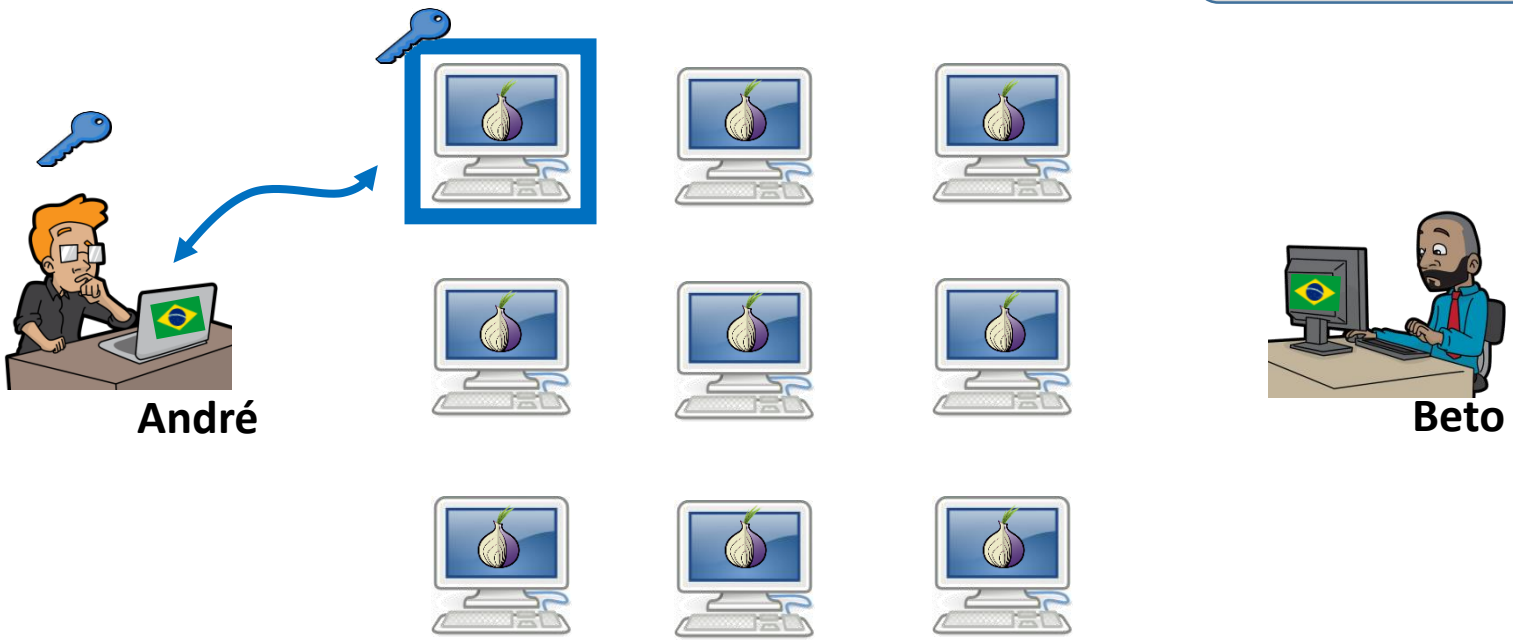


Beto



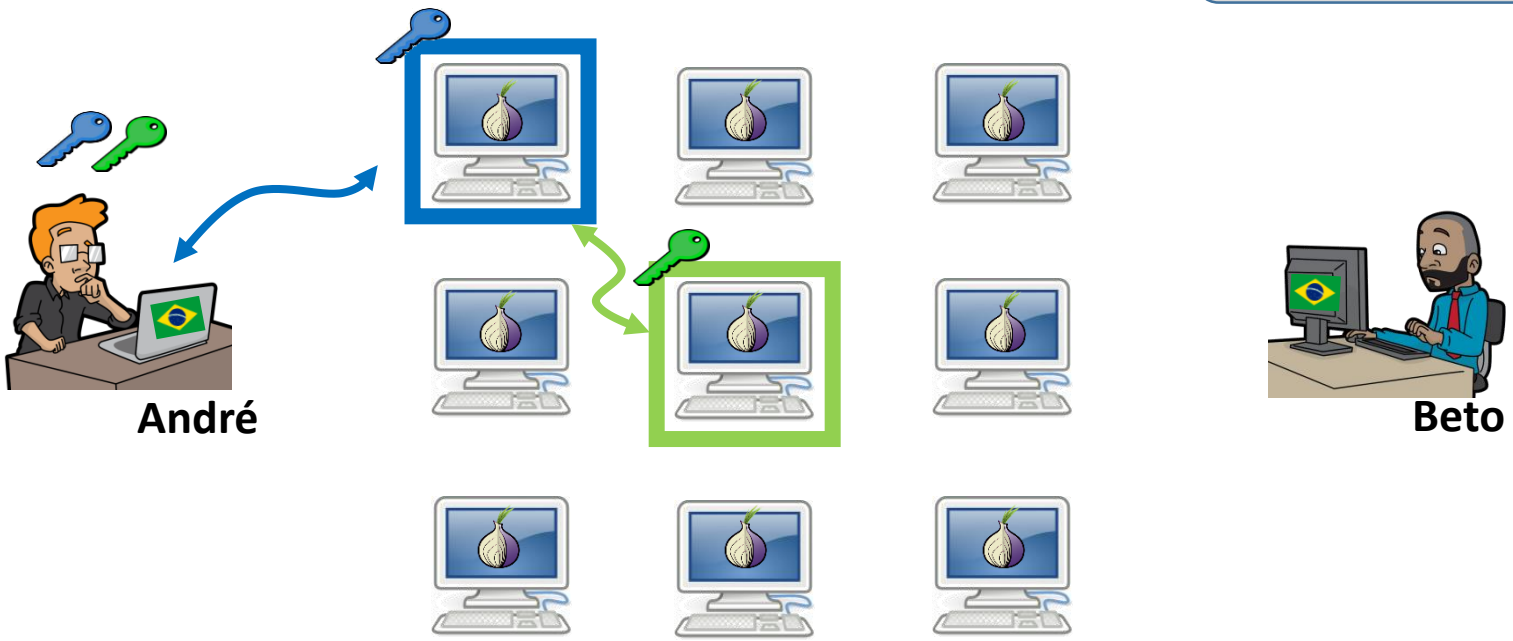
Passo 1: cliente Tor obtém lista de nós Tor em servidor de diretórios

Tor: resumo



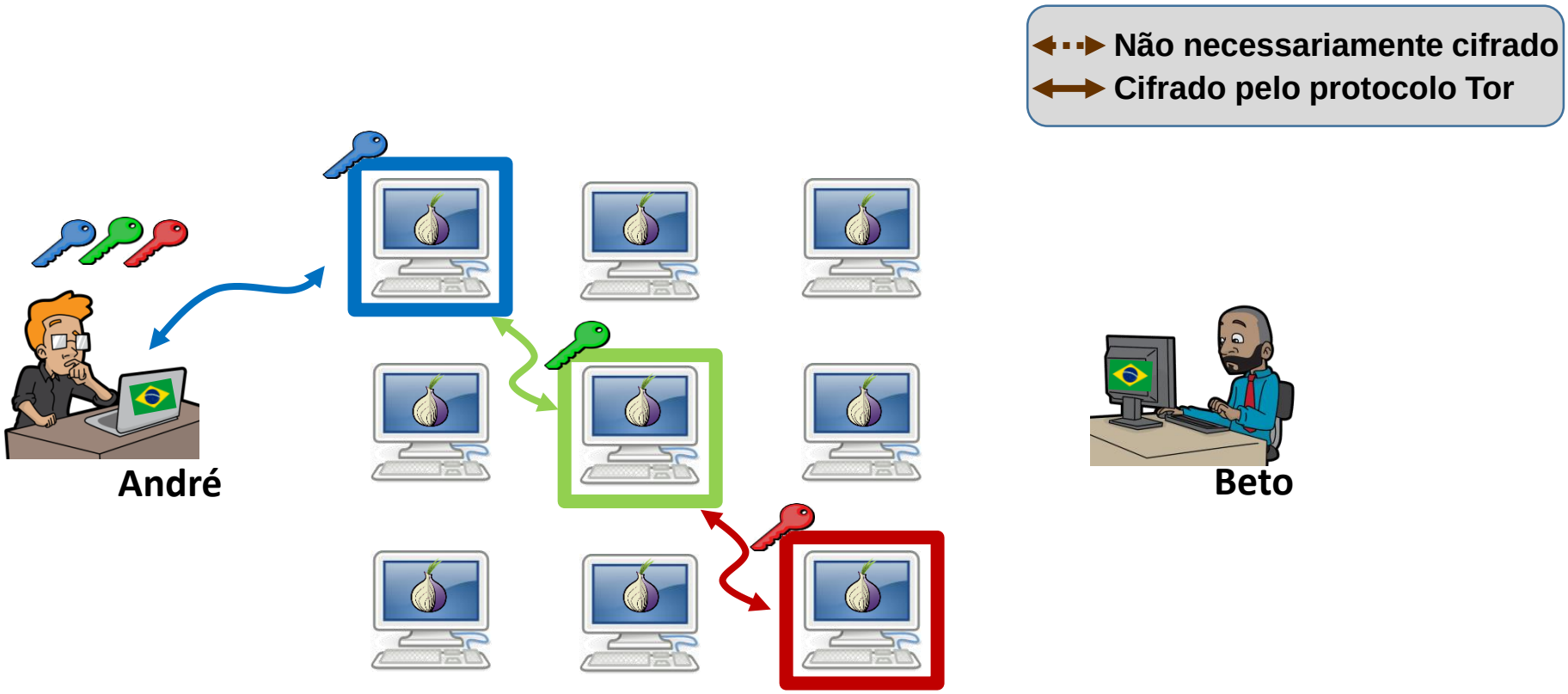
Passo 2: Chave simétrica de sessão com Onion Router #1: (circuito inicial)

Tor: resumo



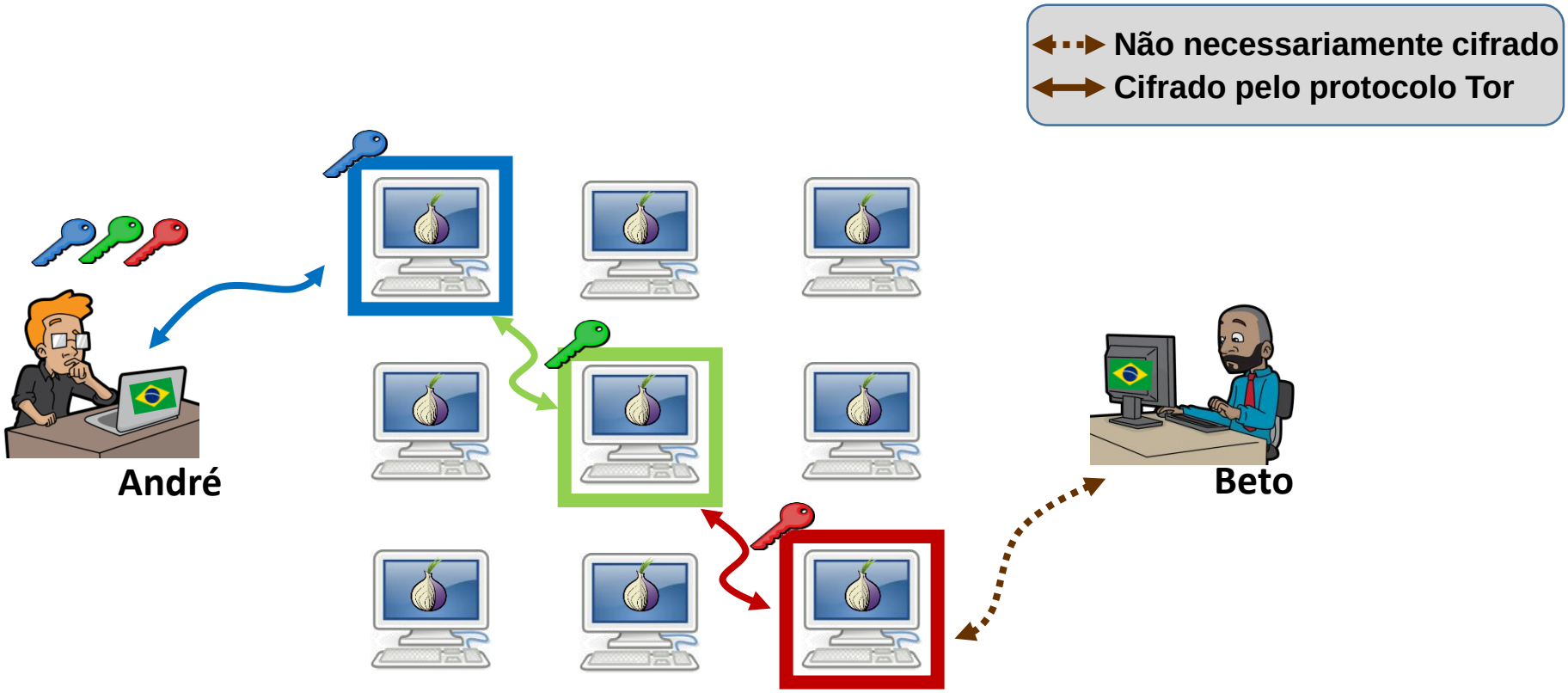
Passo 3: circuito estendido via nova chave simétrica de sessão com Onion Router #2 (tunelamento via router #1)

Tor: resumo

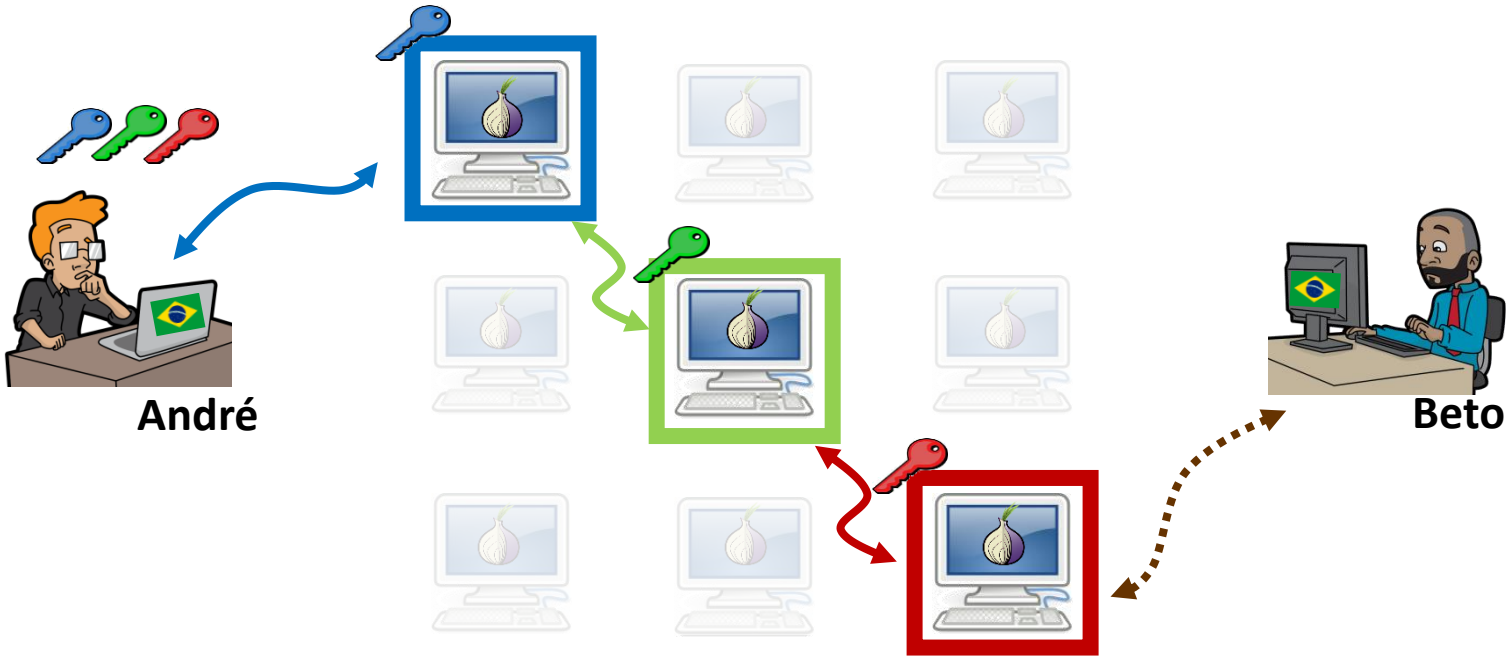


Passo 4: circuito estendido via nova chave simétrica de sessão com Onion Router #3 (tunelamento via router #1 e #2)

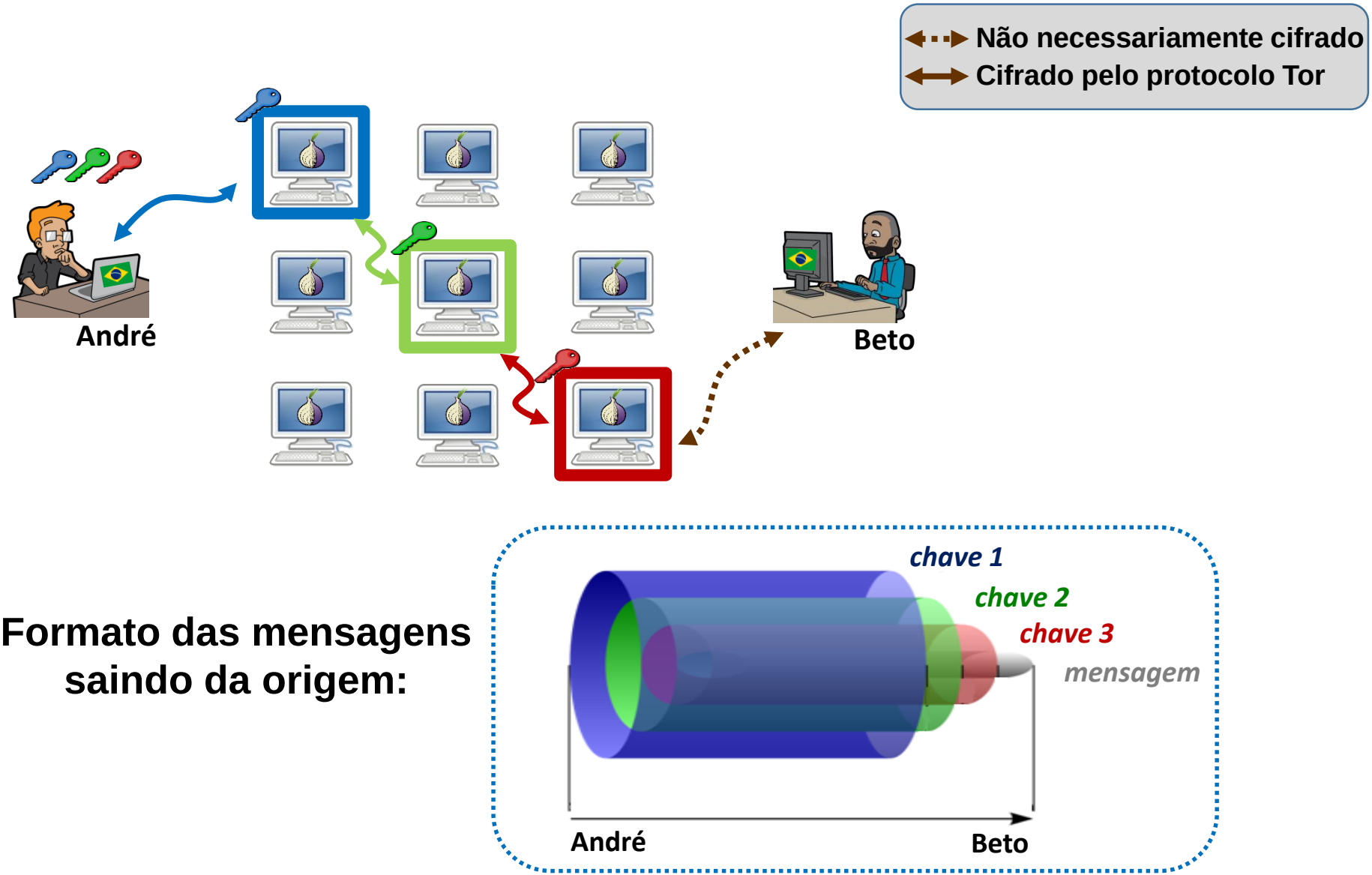
Tor: resumo



Passo 5: Cliente acessa destino final via circuito estabelecido: destino vê apenas Onion Router #3

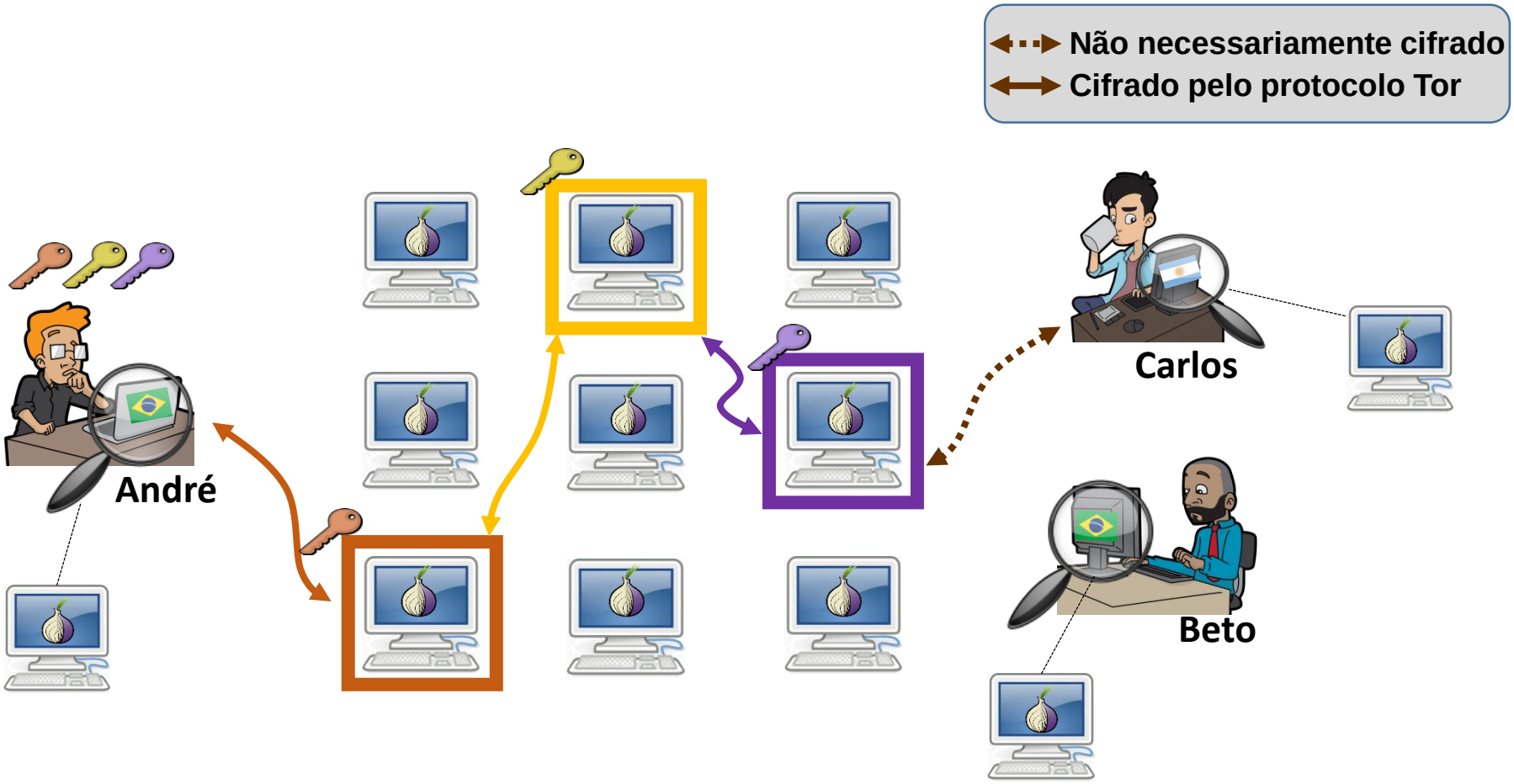

André
Beto

Tor: resumo



Formato das mensagens saindo da origem:

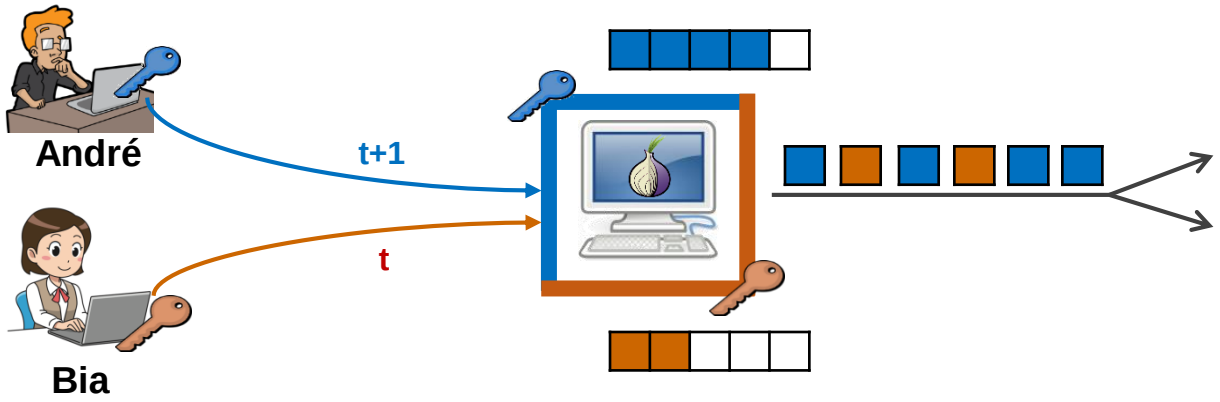
Tor: resumo



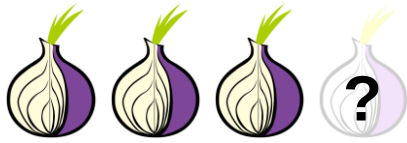
- Novo circuito pode ser estabelecido quando desejado
- Todos os usuários atuam como nós Tor

Tor: características

- **Comutação de células**: privacidade contra escuta
  - Pacotes trafegados têm 512 bytes (header:3, payload:509): previne identificação de fonte pelo **tamanho** das mensagens
  - Algum **embaralhamento interno** nos nós Tor evita deanonimização por **temporização**
    - Objetivo principal é "**equidade**", mas ajuda privacidade
    - Embaralhamento mais robusto **aumentaria latência**



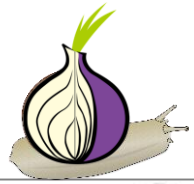
Por que 3 nós e não mais?



- Pergunta: ter mais nós aumenta segurança...?
- Cenário simplificado: 10% da rede comprometida:
  - Qual a chance de pelo menos 1 nó ser malicioso, e derrubar a conexão por não ter controle sobre outros nós relevantes (D)?
- 3 nós: $D = 1-(9/10)^3 = 27\%$
- 4 nós: $D = 1-(9/10)^4 = 34\%$

➔ Mais nós facilita **negação de serviço** por nó malicioso que acredita não conseguir monitorar comunicação...

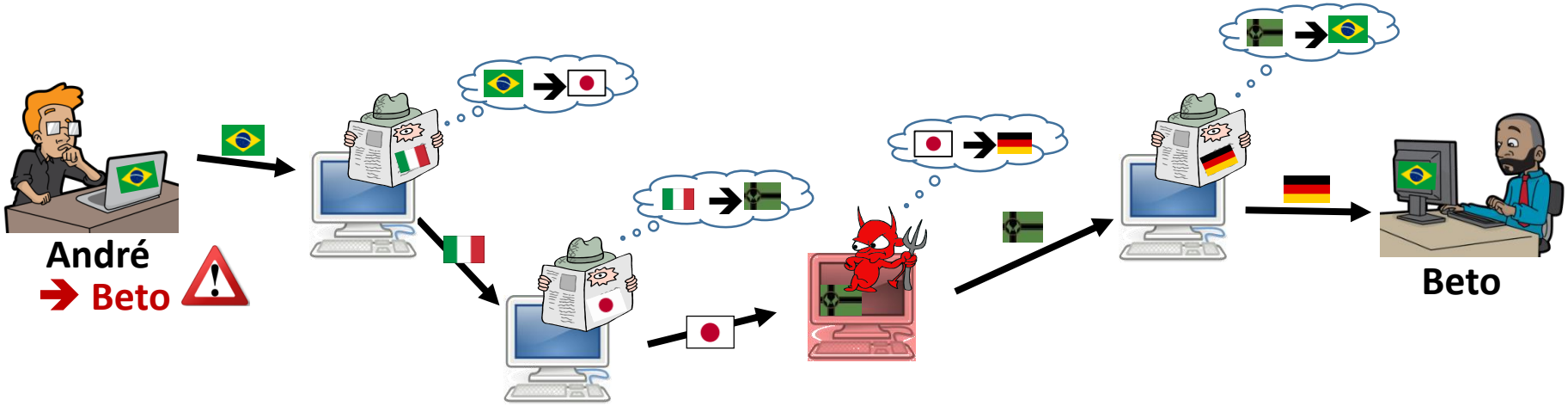
➔ Mais nós também **reduz desempenho** (latência e possibilidade de queda de algum link da conexão)



Tor: características

- "Perfect Forward secrecy"
  - No onion routing, um nó poderia gravar mensagens e depois comprometer a chave privada de todos os nós até o destino
  - No Tor, são criadas chaves de sessão, que são removidas após uso e, assim, não podem ser comprometidas
- Alguns serviços:
  - Diversos fluxos TCP podem **compartilhar** um mesmo circuito
  - Verificação de **integridade no ponto de saída** da rede
  - Nós confiáveis atuam como **servidores de diretório**: listas de roteadores conhecidos assinadas digitalmente
  - Pontos de encontro (rendezvous) e **serviços escondidos**: anonimato dos servidores

Servidores com localização oculta



- Mas e a proteção do IP dos servidores?
  - Origem enxerga IP dos destino!
    - Pode-se usar geolocalização: ataques físicos (ex.: captura)
    - Pode-se usar IP: ataques lógicos (ex.: negação de serviço)
  - Mecanismo extra do Tor: **serviços escondidos (.onion)**

Servidores com localização oculta

- Objetivo: servidor na Internet com as seguintes características:

  - **Disponibilidade**: acessível por qualquer pessoa de qualquer lugar,

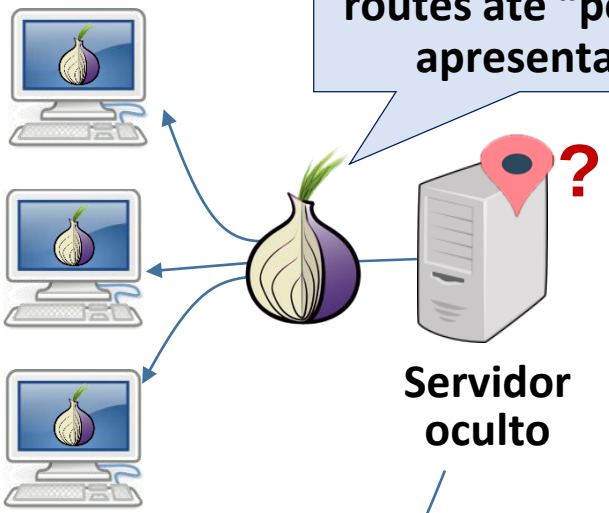
  - **Privacidade**: pessoas que acessam servidor não sabem onde ele está ou quem o controla

- **Resultado**: servidor resistente a censura

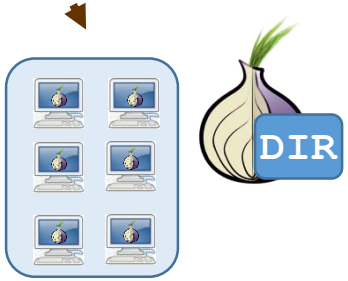
  - Capaz de resistir a ataques de negação de serviço: serviço distribuído, com acesso controlado
  - Resistente a captura física: não se sabe onde está o servidor físico!

Criando um servidores com localização oculta

1) Servidor cria onion routes até “pontos de apresentação”



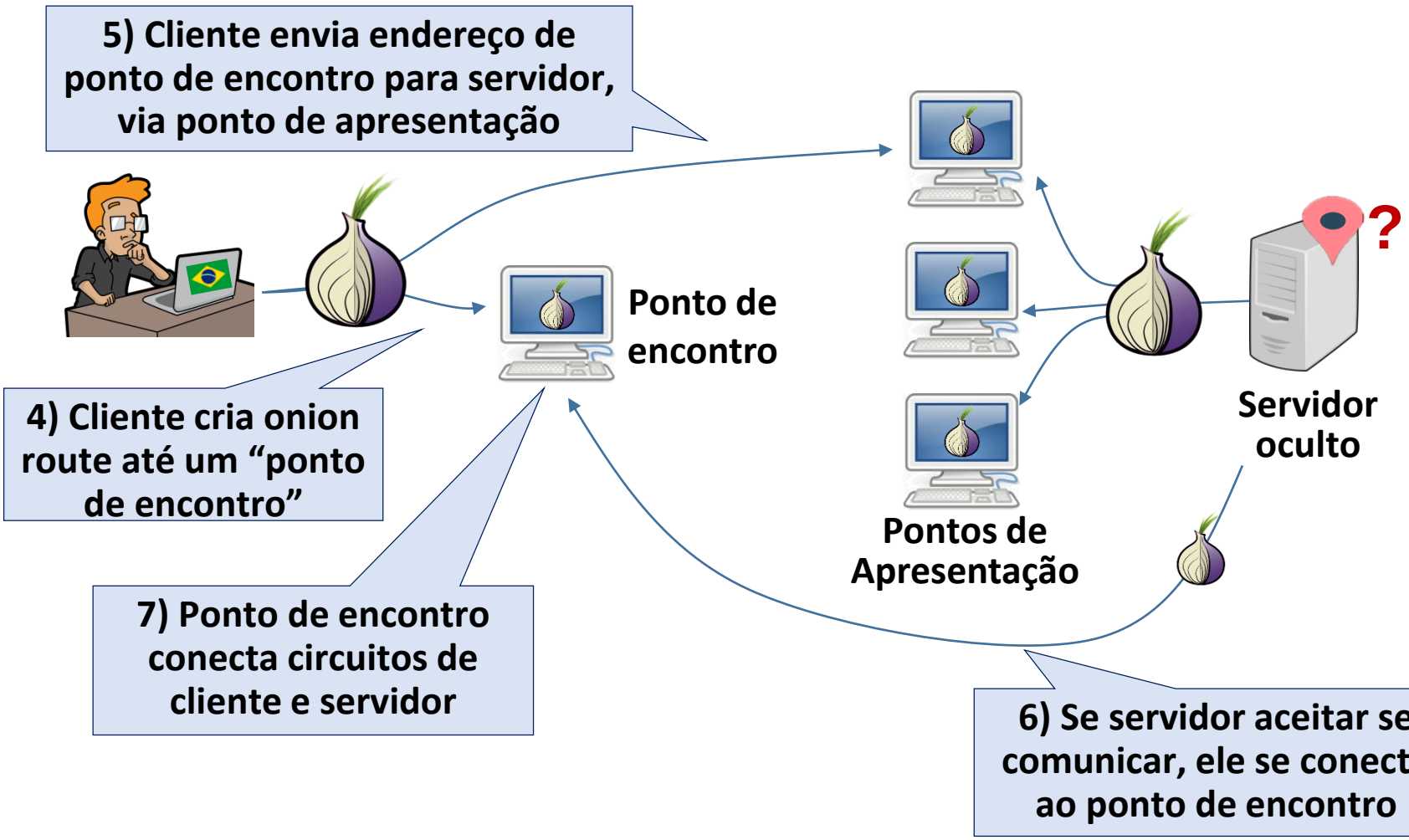
3)Cliente obtém descritor de serviço e endereço de ponto de apresentação



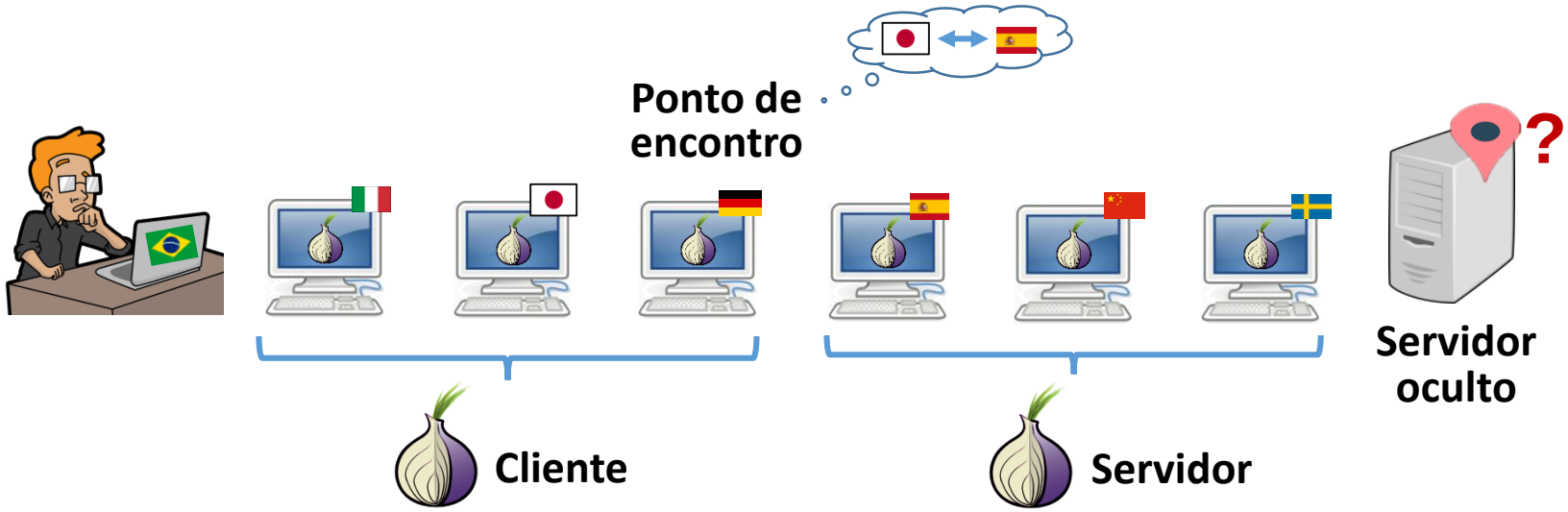
http://3g2upl4pq6kufc4m.onion/

2) Servidor fornece sua descrição e endereços de pontos de apresentação para serviço de diretório

Criando um servidores com localização oculta



Criando um servidores com localização oculta



- Resultado:
  - Servidor não enxerga IP de cliente
  - Cliente não enxerga IP de servidor
  - Nós intermediários não identificam quais são os pontos finais da comunicação

Ataques à rede Tor

- Ataques passivos
  - Não é tão difícil saber se um nó está executando protocolo Tor: o difícil é saber com quem ele está se comunicando
- Ataques ativos
  - DDoS, controle de um nó da rede Tor
- Ataques aos diretórios
  - Destruição ou subversão de servidores de diretório
- Pontos de encontro
  - Ataque a pontos de encontro ou pontos de apresentação

Tor = “Deep Web”?

"Especula-se que a Deep Web [Web profunda] é cerca de 400-500 vezes maior do que a Surface Web [web da superfície]".

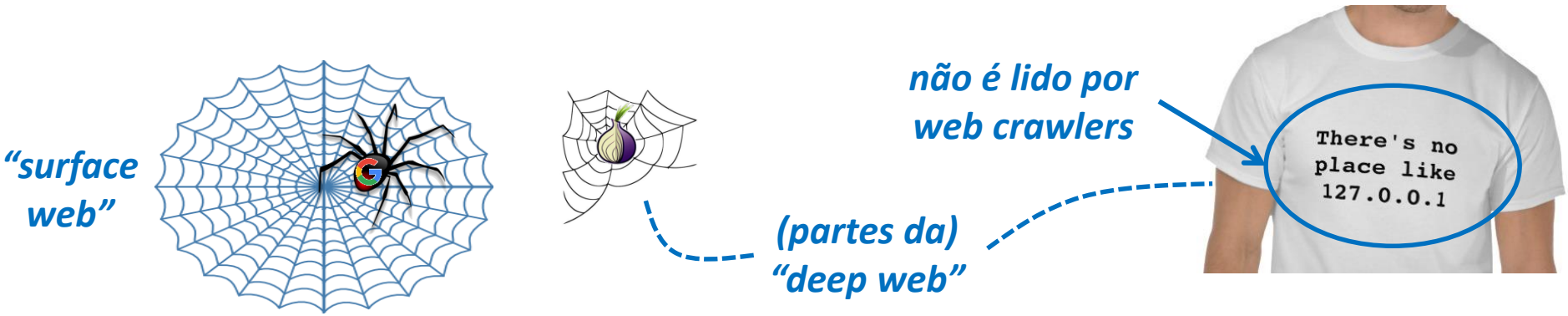
UC Berkeley, 2001

Tor, Web e "Deep Web": Indexação

- **Web Crawling**: **indexação** automática de páginas Web
  - Usado, por exemplo, para construir a base de dados de **sites de busca**
- **Web crawlers** (ou Web spiders): programas de computador que **automatizam indexação**
  - Visitam páginas e indexam texto visível e metadados
  - Seguem hiperlinks encontrados, continuando "navegação" pela Web e descobrindo novos sites
  - Podem executar indefinidamente, identificando modificações em páginas já visitadas.

Tor, Web e "Deep Web": Indexação

- "Deep web": conteúdo web não indexado por máquinas de busca tradicionais
  - Conteúdo **gerado dinamicamente** (ex.: via Javascript)
  - Conteúdo para o qual **não existem links** em sites já indexados
  - Conteúdo em **sites privados ou de acesso restrito**
    - Exigem login ou acesso via canal específico (ex.: redes Tor ou Freenet)
  - Texto embutido em **arquivos multimídia**

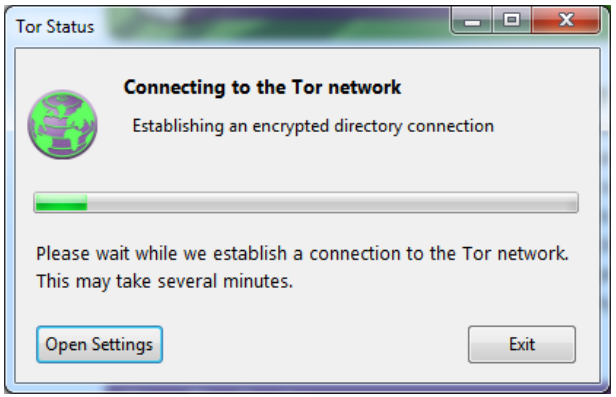


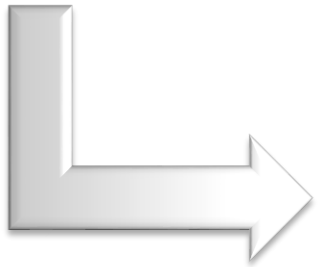
Tor, Web e "Deep Web": Indexação

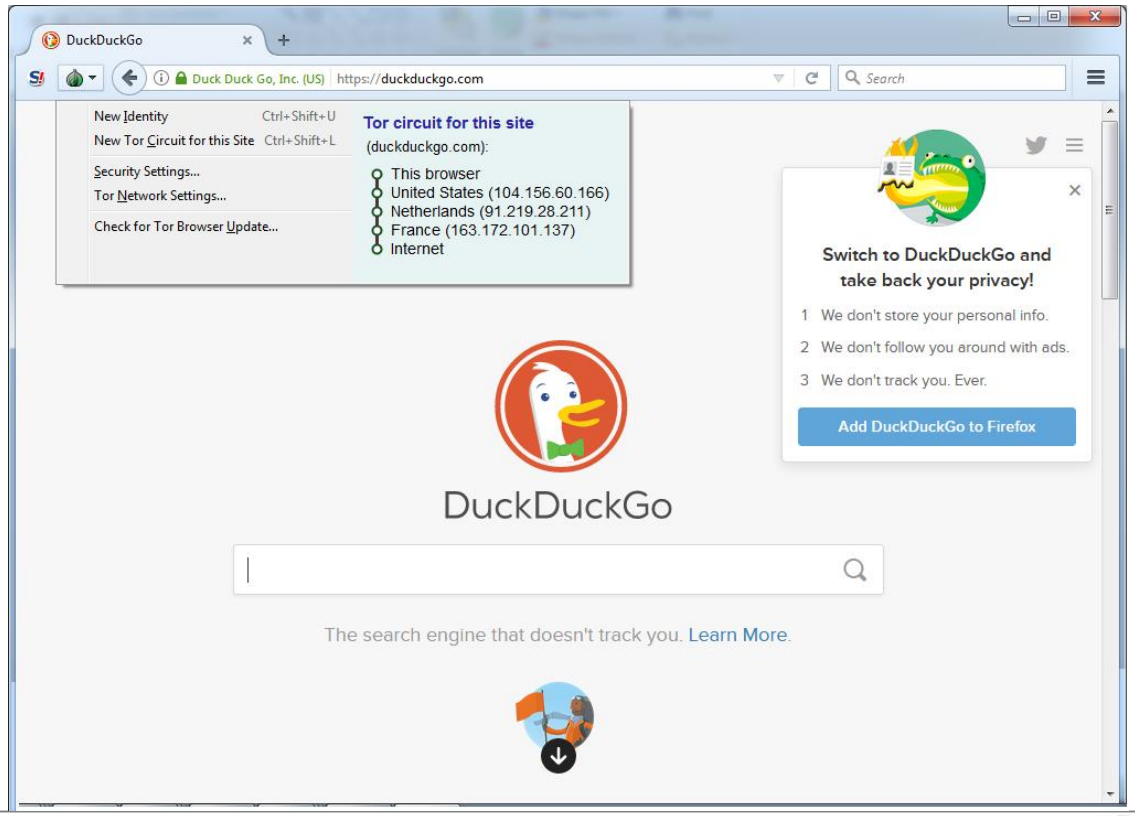
Termo às vezes usado (de modo simplista) como "conteúdo ilegal"

Tor Browser: teste você mesm@!

- Página oficial: https://www.torproject.org/
  - Ex. de "hidden wiki": https://thehiddenwiki.org/
  - http://zqktlwi4fecvo6ri.onion/wiki/index.php/Main_Page







Blockchain, Criptomoedas & Tecnologias Descentralizadas

Tecnologias descentralizadas:
Tor e Privacidade

Prof. Dr. Marcos A. Simplicio Jr. – mjunior@larc.usp.br
Escola Politécnica, Universidade de São Paulo

Apoio financeiro: d24n.org <https://d24n.org>
ripple impact | university blockchain research initiative