Vol.2015-DPS-163 No.26
Vol.2015-MBL-75 No.26
2015/5/29

プレフィックスの階層性に基づいたCache Pollution Attackへの対策の考察

神本　崇史[†1]　梅田　沙也華[†1]　大畑　百合[†1]　重野 寛[†1]

概要：Named Data Networking（NDN）において，Cache Pollution Attack の問題が指摘されている．この攻撃により，正常なユーザから要求される確率が低い人気のないコンテンツで Content Store（CS）と呼ばれるキャッシュが汚染され，正常なユーザのコンテンツ取得時間の遅延が発生する．NDN における Cache Pollution Attack に対する既存手法としてコンテンツ名に着目した攻撃検知手法がある．これは，攻撃の検知のみを行い，その影響の抑制までは対象としていない．本稿では，NDN における Cache Pollution Attack への対策として，コンテンツ名の持つプレフィックスの階層性に着目した手法 RMCP を提案する．RMCP は攻撃を検知した後，その影響の抑制を行う手法である．攻撃により NDN ルータの CS に人気のないコンテンツが保存されるため，正常なユーザのキャッシュヒット率が低下する．そのため，RMCP はキャッシュヒット率低下を抑制することを目的とする．具体的には，攻撃に利用されているコンテンツのプレフィックスを階層性に基づいて識別してブラックリストに格納し，ブラックリストにあるデータを CS に保存しないように保護する．これによって，正常なユーザのキャッシュヒット率低下を抑制し，コンテンツ取得時間の遅延発生を軽減する．また，シミュレーションを用いて既存手法と RMCP の比較評価を行い， 攻撃による影響における RMCP の有意性を確認した．

キーワード：Named Data Networking; Cache pollution attack; content name prefix

1. はじめに

近年，IP ネットワークに代わるネットワークアーキテクチャとして Named Data Networking（NDN）が研究されている [1,2]．NDN はコンテンツ指向アーキテクチャと呼ばれるもので，ユーザはコンテンツの場所を指定するのではなく，コンテンツの名前を指定することによって通信を行う．このようなアーキテクチャはコンテンツ取得を効率的に行うことができるという利点を持っている．NDN における各ルータは Content Store（CS）というキャッシング機構を持っている．ユーザはコンテンツを保持しているサーバ以外にも，コンテンツを CS に保存しているルータからコンテンツを取得することができ，取得時間の短縮を行うことが出来る．NDN ではパケットが署名を持ち，その構造から IP において問題とされていた多くの攻撃に対して耐性をもつ．

しかし，NDN では Cache pollution attack という CS に対する攻撃が問題とされている [3]．この攻撃は正常なユーザから要求される確率が低い人気のないコンテンツで CS を汚染し，正常ユーザが CS を有効活用することを妨げる攻撃である．CS が汚染されると正常なユーザがコンテンツを取得する時間に遅延が発生する．NDN における Cache Pollution Attack に対する既存手法としてコンテンツ名に着目した攻撃検知手法の研究がある [4]．これは，攻撃の検知のみを行い，影響の抑制までは対象としていない．

そこで，本稿ではこの攻撃への対策手法である RMCP を提案する．RMCP はコンテンツ名の持つプレフィックスの階層性に着目した手法であり，攻撃の検知を行った後にその影響を抑制するものである．Cache pollution attack によって NDN ルータにおける正常なユーザのキャッシュヒット率が低下することが知られているため，RMCP はこのキャッシュヒット率低下を抑制することが目標である．RMCP の具体的な動作として，まず攻撃に利用されているコンテンツのプレフィックスを階層性に基づいて識別してブラックリストに格納する．その後，ブラックリストにあるデータを用いて CS を汚染状態から回復させ，攻撃に利用されているコンテンツを保存しないように保護する．このプレフィックスの識別を用いた対策によって，正常なユーザのキャッシュヒット率低下を抑制する．また，本稿では

[†1] 現在，慶應義塾大学大学院理工学研究科
Presently with Graduate School of Science and Technology, Keio University

シミュレーションを用いて既存の攻撃検知手法と RMCP の比較評価を行い， 攻撃による影響における RMCP の有意性を確認した．

本稿の構成は以下のとおりである．第 2 章では NDN の概要や Cache pollution attack とその関連研究について紹介する．第 3 章では提案手法である RMCP について説明する．第 4 章ではシミュレーション結果から RMCP の効果を検証する．最後に，第 5 章で結論を述べる．

2. NDN の概要と関連研究

本節では Named Data Networking の概要について紹介する．次に NDN において問題とされている Cache pollution attack について説明し，その対策の関連研究を紹介する．

2.1 Named Data Networking

Named Data Networking（NDN）は現在の IP ネットワークに代わるアーキテクチャとして検討されている [2]．IP では IP アドレスを指定して通信を行っているため，通信を行うためにはコンテンツがどこにあるかを知る必要がある．一方で，NDN ではコンテンツの場所に焦点を置かず，コンテンツ名を指定することで通信を行うことが出来るので，コンテンツ取得の効率化が可能である．NDN におけるパケットは二種類あり，それぞれ要求パケットを Interest パケット，それに対する応答パケットを Data パケットである．各 NDN ルータは Forwarding Information Base（FIB），Pending Interest Table（PIT），Content Store（CS）という三つの機構を持っている．FIB は Interest を転送する際に用いる転送表で，これによって Interest を転送する方向を決定する．PIT は Data を転送する際に参照する表で，Data に含まれるコンテンツを要求したユーザの方向へ Data が転送される．CS はキャッシング機構で，各ルータはキャッシングポリシーに従って CS にコンテンツを保存する．ここで，キャッシングポリシーとはそのコンテンツをキャッシュに保存するかを決定する方針のことである．NDN では各ルータがキャッシュを保持することでコンテンツを分散させ，ユーザは効率的にコンテンツを取得することが可能となる．

NDN におけるコンテンツ名は「/Photos/cont1/k3JrodBs/1/1」のように，「/」と文字列の連続で表現される．ここで，「/」によって区切られた各要素を component と呼ぶ．また，後ろから数個の component を除いた残りの部分をプレフィックスと呼ぶ．先の例では，最短プレフィックスは「/Photos/」となり，最長では「/Photos/cont1/k3JrodBs/1/」となる．コンテンツ名は木構造のような階層構造を持つ．この木構造により，コンテンツ名の中で先頭に近い位置にある component ほど木構造のルートに近いため，多くのコンテンツ名に含まれる．つまり，一つのプレフィックスを指定することは，一般に複数のコンテンツ名を指定することと同義である．サーバは自身のもつコンテンツのプレフィックスをルータに広告することで，各ルータにおいて FIB が設定される．

2.2 Cache pollution attack

Cache pollution attack は正常なユーザから要求される確率の低い人気の無いコンテンツで CS を汚染し，正常なユーザのコンテンツ取得時間に遅延を発生させる攻撃である．Cache pollution attack は IP ネットワークにおいても存在していたが，IP ネットワークにおいては全てのノードがキャッシュを保持しているわけではない．一方で，NDN では全てのノードが CS というキャッシュを保持しているため，Cache pollution attack が大きな問題となっている．Cache pollution attack は大きく二種類に大別される [3]．一方は Locality-disruption 攻撃と呼ばれ，多数の人気の無いコンテンツを幅広く要求することで CS に人気の無いコンテンツを保存させる攻撃である．この攻撃で要求される各コンテンツの要求量は少なく，ルータは要求量が少ないコンテンツを CS に保存せざるを得ない状況になる．これによってキャッシュが近傍に存在するノードの人気を反映させる性質であるローカル性が損なわれる．もう一方は False-locality 攻撃と呼ばれ，小数の人気の無いコンテンツを選択して集中的に要求することで CS に人気の無いコンテンツを保存させる攻撃である．この攻撃で要求される各コンテンツの要求量は多く，ルータは正常なユーザから人気のあるコンテンツと攻撃によって要求されている人気の無いコンテンツの見分けができなくなり，誤って人気の無いコンテンツを保存する．これによってキャッシュが誤ったローカル性を持つことになる．どちらも攻撃方法に違いはあるが，ルータの CS に人気の無いコンテンツを保存させ，結果的に正常なユーザのコンテンツ取得に遅延を発生させる点は同じである．

Cache pollution attack による影響の大きさはキャッシュ置換方式に依存する [4]．キャッシュ置換方式とは，キャッシュがあふれた場合にどのコンテンツから削除するかを決定する方式のことである [5–7]．代表的なものに Least recently used（LRU）方式と Least Frequently Used（LFU）方式の二種類がある．LRU は最後の要求された時間が最も古いコンテンツから削除するものであり，LFU は最も要求された頻度が低いものから削除するものである．攻撃で要求される各コンテンツの要求量の特徴から，LFU 方式は Locality-disruption 攻撃の影響を抑制することが可能である．一方で，LRU 方式はどちらの攻撃に対しても有効ではない．キャッシュ置換方式だけでは Cache pollution attack の影響を抑制することはできないため，影響抑制のためには他の手法が必要となる．次の節では，Cache pollution attack に対する関連研究について紹介する．

2.3 Cache pollution attack に対する関連研究

IP における Cache pollution attack の対策も検討されている [3,8]．これらの手法の多くは IP アドレスを用いて攻撃者を識別し，攻撃の対策を行っている．しかし，NDN では IP アドレスのようなユーザを識別する指標がないため，これらの手法を NDN に適用することはできない．以下では，NDN における Cache pollution attack の関連研究を紹介する．

要求のランダム性に着目し，Cache pollution attack を検知する手法がある [9]．本稿ではこの手法を Randomness Check と呼ぶ．この手法では，正常なユーザからの要求には一定のランダム性があるという考えに基づき，要求パケットを集計してランダム性を計算する．このランダム性が閾値を下回った場合に攻撃が行われていると判断する．この手法は高精度の検知を実現したものであり，Locality-disruption 攻撃に対して有効である．しかし，高計算量で検知のみを行う手法であり，False-locality 攻撃に対して有効ではない．

各コンテンツの要求回数に着目し，Cache pollution attack の影響を抑制する手法として CacheShield という手法がある [10]．この手法は各コンテンツ毎に要求された回数を集計し，要求回数が一定値以上になるまでコンテンツを保存しないようにする手法である．CacheShield は Locality-disruption 攻撃に対して有効で，攻撃の影響を抑制するだけでなく，CacheShield がない場合よりも正常なユーザのキャッシュヒット率が向上する．しかし，False-locality 攻撃に対しては有効ではなく，さらに要求回数が一定値を超えるまで時間がかかるため，人気の反映が遅くなる．

各コンテンツの要求率に着目し，Cache pollution attack を検知する手法がある（本稿ではこの手法を Lightweight attack detection と呼ぶ）[4]．この手法は予想の要求率と実際の要求率の差である要求率変化量をコンテンツ毎に算出し，その合計値を調べて正常なユーザの人気から予測される値とどのくらいの差があるかを調べる．コンテンツ i の要求率 $p(i)$ は次の式で算出される．

$$p(i) = \frac{n_r(i)}{\sum_{j \in S} n_r(j)} \tag{1}$$

ここで，S は要求率を算出するコンテンツの集合をあらわし，$n_r(i)$ はコンテンツ i の要求回数である．次に，予想の要求率と実際の要求率の差であるコンテンツ毎の要求率変化量 δ_i を次式によって算出する．

$$\delta_i = (\frac{n_r^m(i)}{N_r^m} - p(i)) \tag{2}$$

ここで，m が過去の測定で得た値であることを示し，N_r^m は過去の測定における要求回数の合計値を意味する．つまり，$\frac{n_r^m(i)}{N_r^m} - p(i)$ とは，過去のある時点におけるコンテンツ i の要求率である．この δ_i を合計して，差の合計値 δ_m を算出する．予想との差の合計値 δ_m が閾値を上回ると攻撃が行われていると判断する．この手法は Locality-disruption 攻撃と False-locality 攻撃の両方を検知することが出来る．しかし，検知のみを行っているため，攻撃の影響を抑制することはない．

Randomness check や CacheShield のように，NDN における Cache pollution attack の関連研究の多くは Locality-disruption 攻撃に焦点をおいている．False-locality 攻撃に焦点を当てた研究として Lightweight attack detection があり，これは Randomness Check と CacheShield の後に考案された手法であるために低計算量と低情報量を実現した手法であるが，攻撃の検知を行うのみで，攻撃の影響を抑制することはない．そこで，低計算量と低情報量を実現し，False-locality 攻撃の影響を抑制する手法が必要である．

3. 提案手法 RMCP

本節では，本稿の提案手法である RMCP（Resistant Method against cache pollution attack based on hierarchy of Content name Prefix in Named Data Networking）について詳細に説明する．

RMCP はコンテンツ名の持つプレフィックスの階層性に着目した手法であり，攻撃の検知を行った後にその影響を抑制するものである．ルータにおいて攻撃が検知されると，そのルータで RMCP が実行されてそのルータにおける攻撃の影響を抑制する．Cache pollution attack によって NDN ルータにおける正常なユーザのキャッシュヒット率が低下することが知られているため，RMCP はこのキャッシュヒット率低下を抑制することが目標である．RMCP は Cache pollution attack の中でも特に False-locality 攻撃の影響を抑制する手法である．Locality-disruption 攻撃によって要求されるコンテンツのプレフィックス数は正常ユーザの要求するコンテンツのプレフィックス数より多く，プレフィックスの識別が難しい．一方で，False-locality 攻撃ではそれらの数が大きく異なることはないため，プレフィックスの識別がより簡単に行える．このことから，RMCP は False-locality 攻撃の影響を抑制することが可能である．

False-locality 攻撃の影響抑制を低計算量，低情報量で行うために，この手法における攻撃検知の段階は，Lightweight attack detection の考えを応用したものを用いる．この手法の考え方を用いることで低計算量と低情報量の利点を得ることができる．しかし，検知のために情報量が減少しており，そのまま攻撃の影響を抑制することに応用することは難しい．そこで，Lightweight attack detection の考え方を応用しながらも別の方法によって情報量の削減を行う．具体的には，Lightweight attack detection では要求率を観測するコンテンツを制限していたが，RMCP ではそのよう

Vol.2015-DPS-163 No.26
Vol.2015-MBL-75 No.26
2015/5/29

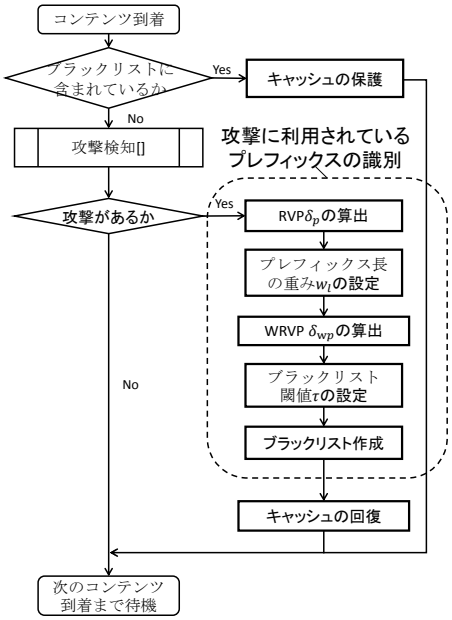

図 1 RMCP の動作の流れ

な制限を行わず，コンテンツ毎ではなくプレフィックス毎に要求率をまとめることで情報量の削減を行う．

RMCP の動作の流れを図 1 に示す．RMCP の動作は以下の 3 つのステップから構成される．

- 攻撃に利用されているプレフィックスの識別
- キャッシュの回復
- キャッシュの保護

RMCP では攻撃に利用されているコンテンツのプレフィックスを階層性に基づいて識別し，ブラックリストに格納する．このプレフィックスの識別には後述の Weighted Request Variation pre Prefix（WRVP）という値を用いる．WRVP は攻撃者から利用されている可能性の高さを示す値であり，この値が大きいプレフィックスほど攻撃者に利用されている可能性が高い．各プレフィックスの WRVP とブラックリスト閾値を比較することにより，ブラックリストを作成する．その後，ブラックリストに含まれるプレフィックスを含むコンテンツを CS から削除することで CS を汚染状態から回復させ，攻撃に利用されているコンテンツを保存しないように保護する．これらの動作によって，Cache pollution attack による正常なユーザのキャッシュヒット率低下を抑制する．

以下では，RMCP の各ステップについて詳細に述べる．

3.1 攻撃に利用されているプレフィックスの識別

RMCP では，攻撃が検知された後にまず攻撃者に利用されているプレフィックスの識別を行う．プレフィックスの識別では，まずプレフィックス毎の要求率変化量（RVP，

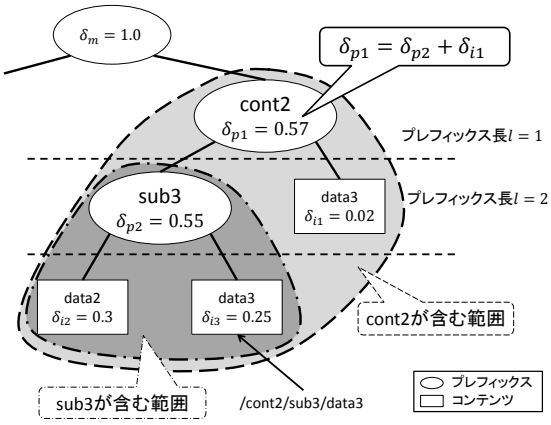

図 2 プレフィックスの階層構造と RVP

Request rate Variation per Prefix）δ_p を算出する．これはプレフィックス毎の攻撃に利用されている可能性を示す値であるが，プレフィックスの階層構造により RVP によるプレフィックスの識別は誤検知しやすいという問題がある．そこで，RVP にプレフィックスの長さに対する重みを反映させた要求率変化量（WRVP，Weighted RVP）を算出する．WRVP はプレフィックスの階層構造の影響を低減させた値であり，この値を用いて識別を行い，ブラックリストを作成する．

攻撃に利用されているプレフィックスの識別の最初の段階として，RVPδ_p を算出する式は次の通りである．

$$\delta_p = \sum_{i \in S_p} \delta_i, \qquad (0 \leq \delta_p \leq 1) \tag{3}$$

δ_i は式 2 により攻撃検知時に算出される値で，S_p は同じプレフィックスを持つコンテンツの集合を意味する．RVPδ_p は，$0 \leq \delta_p \leq 1$ の範囲を持つように要求率変化量の総和によって正規化される．Lightweight attack detection ではこのコンテンツ毎の要求率変化量 δ_i を算出するコンテンツを制限することにより情報量を削減していた．この制限を行うと攻撃の影響抑制が難しくなるため，RMCP では検知段階でこの制限を行わず，プレフィックスの識別を行うときにプレフィックス毎の値にまとめることで情報量の削減を行う．プレフィックスは階層構造を持つため，木構造の根に近いプレフィックスほど大きな RVP を持つことになる．

図 2 にプレフィックスの階層構造と RVP の関係の例を示す．「/cont2/」というプレフィックスが含む範囲は「/sub3/」というプレフィックスが含む範囲より大きい．RVPδ_p はそのプレフィックスが含む範囲内にあるコンテンツ毎の要求率変化量 δ_i の総和である．このことから，根に近いプレフィックスほど多くのコンテンツを含むので，RVPδ_p も大きくなり，「/cont2/」の δ_{p1} は「/sub3/」の δ_{p1} より大きくなる．よって，RVP のみを用いてプレフィックスの識別を行った場合，多くのコンテンツを含むプレフィックスが攻撃に利用されていると判断されてブラッ

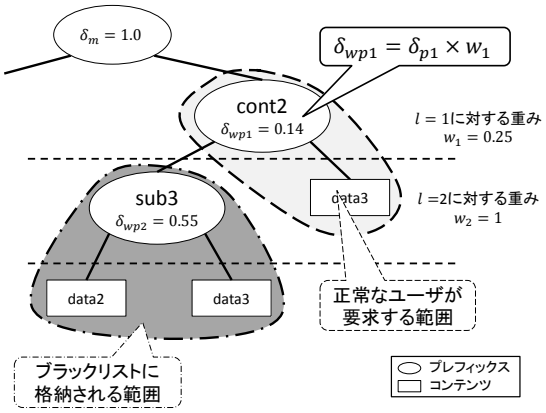

図 3 プレフィックスの階層構造と WRVP

クリストに格納される可能性が高い．これにより，正常なユーザが要求する人気のあるコンテンツのプレフィックスがブラックリストに含まれる危険性が生じる．例えば，図 2 において「/cont2/sub3/」というプレフィックスをもつコンテンツを攻撃者が要求しており，他のコンテンツを正常なユーザが要求しているとする．RVP の値は「/sub3/」より「/cont2/」の方が大きいため，ブラックリストに格納されるのは「/cont2/」となる．すると，正常なユーザが要求している「/cont2/data3」というコンテンツもブラックリストに格納されるという問題が発生する．

そこで，根から離れたプレフィックスほど大きな値を持つような補正を行う必要がある．この補正にはプレフィックス長 l を用いる．プレフィックス長 l とは，プレフィックスの含む component の数のことである．例えば，「/Photos/cont1/k3JrodBs/1/」というプレフィックスは $l = 4$ となる．このプレフィックス長 l は根から遠ざかるほど大きな値を持つ．

プレフィックス長 l に応じた重み w_l を RVP にかけることによって補正を行い，重みを反映させた WRVP を用いてプレフィックスの識別を行う．このプレフィックス長を反映させるための重み w_l はプレフィックス長 l が大きいほど大きな値を持つように設定する．WRVPδ_{wp} を算出する式は次の通りである．

$$\delta_{wp} = \delta_p \times w_l. \qquad (0 \leq w \leq 1,\ 0 \leq \delta_{wp} \leq 1) \qquad (4)$$

この値を用いることにより，プレフィックス長に関わらず攻撃に利用されている可能性によってプレフィックスの識別を行うことが可能である．

図 3 にプレフィックスの階層構造と WRVP の例を示す．プレフィックス長 $l = 1$ に対する重みを $w_1 = 0.25$ とし，プレフィックス長 $l = 2$ に対する重みを $w_2 = 1$ とする．この重みと RVP を用いて書くプレフィックスの WRVP を算出すると，「/cont2/」の値は 0.14，「/sub3/」の値は 0.55 となる．RVP を用いて識別を行うと，木構造の根に近いプレフィックスほど大きな値を持っていたが，WRVP を用いると根から離れたプレフィックスも大きな値を持つことができる．先ほどの例と同様に攻撃者が「/cont2/sub3/」を利用していた場合，「/sub3/」の WRVP は「/cont2/」の WRVP より大きな値を持つために正しくブラックリストを作成することができる．このようにして，プレフィックスの階層性に基づき，利用することで効果的な識別が可能となる．

WRVP を算出すると，次に WRVP が高いプレフィックスをブラックリスト候補とする．このブラックリスト候補の WRVP を基準としてブラックリスト閾値 τ を設定する．RMCP が実行されるのは攻撃が検知された後であるので，少なくとも 1 つのプレフィックスがブラックリストに含まれるように閾値 τ を設定する．このように閾値を設定することで，状況に応じて動的に閾値が設定されるようにする．閾値 τ の設定が終わると，各プレフィックスの WRVP と閾値を比較し，WRVP が閾値を超えたプレフィックスをブラックリストに格納する．

3.2 キャッシュの回復

攻撃に利用されているプレフィックスの識別によってブラックリストを作成すると，WRVP はキャッシュの回復段階に進む．攻撃が検知された時点で既に CS は人気のないコンテンツが保存されることによって汚染されている．キャッシュの回復では，CS を汚染されていない状態に戻す．各ルータは CS の中に含まれているコンテンツのプレフィックスがブラックリストに含まれていないかどうか調べる．もしブラックリストに含まれているプレフィックスをもつコンテンツがあると，そのコンテンツを CS から削除する．これによって，CS から攻撃に利用されていたコンテンツを削除し，汚染されていない状態に戻す．

3.3 キャッシュの保護

キャッシュの保護によって CS を汚染が無い状態に戻した後，キャッシュが再び汚染されないように保護する．ルータに新たなコンテンツが到着すると，そのコンテンツのプレフィックスがブラックリストに含まれていないか調べる．もしブラックリストに含まれていた場合，そのコンテンツを CS に保存せずに転送を行う．これによって，CS に攻撃者が利用している可能性の高いコンテンツによって CS が再び汚染されるのを防ぐ．

4. シミュレーション評価

RMCP の Cache pollution attack に対する有効性を検証するために，シミュレーションによる評価を行った．比較対象として，攻撃の影響を抑制せずに攻撃の検知のみを行う Lightweight attack detection を用いた．また，評価項目はルータにおける正常なユーザからの要求のキャッシュヒット率である．これは正常なユーザの要求数に対して

表 1　共通シミュレーション条件

ネットワークシミュレータ	ns-3.20
NDN モジュール	ndnSIM
シミュレーション時間	7200 sec
トポロジー	XC，DFN
キャッシュ置換方式	LRU
CS 容量	コンテンツ全体の 1%
攻撃者の要求コンテンツ数	CS 容量と同程度
攻撃者の要求パケット量	毎秒正常なユーザの合計と同量

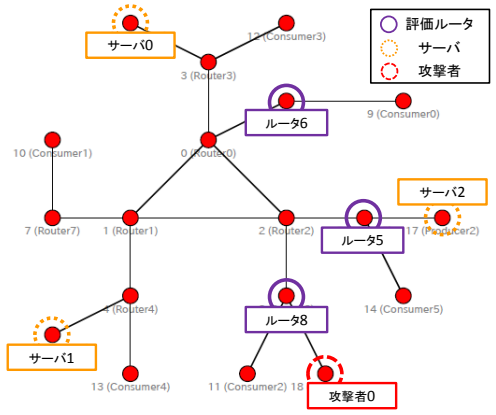

図 4　XC トポロジ

そのルータからコンテンツを返信した割合で計算される。Cache pollution attack の目的はこの値を下げることであり，RMCP は攻撃による低下を抑制することが目標である．以下ではシミュレーション環境について説明した後，結果を示す．

4.1　シミュレーション環境

本評価ではネットワークシミュレータとして ns-3 [11] と，NDN の機能として ns-3 用の NDN モジュールである ndnSIM を用いた [12]．既存手法と RMCP の共通シミュレーション条件を表 1 に示す．シミュレーション条件の多くは Lightweight attack detection で用いられている条件に従っている．シミュレーション時間は 2 時間で，前半の 1 時間は正常なユーザのみが要求を行い，各 NDN ルータに正常なユーザの人気を反映させる．後半の 1 時間では正常なユーザと攻撃者の両方が要求を行う．正常なユーザは Zipf の法則に従ってコンテンツを要求する．つまり，正常なユーザは人気の高いコンテンツほど高確率で要求する．一方，攻撃者は人気のないコンテンツを選択して一様分布に従い要求を行う False-locality 攻撃を想定している．

使用したトポロジは XC トポロジ [10]，DFN トポロジ [13] の 2 つである．XC トポロジの概形を図 4 に，DFN トポロジの概形を図 5 に示す。また，各トポロジにおけるノード数を表 2 に示す．XC トポロジは攻撃者との距離が異なること以外は同じ条件を持つノードを配置したトポロジである．このため，攻撃者の影響を確認することに適し

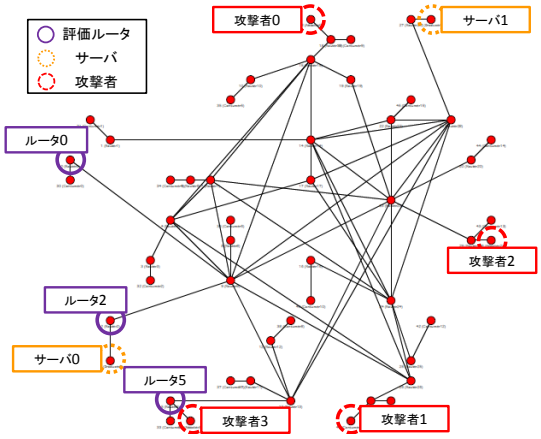

図 5　DFN トポロジ

表 2　各トポロジのノード数

トポロジ名	XC	DFN
NDN ルータ数	9	30
正常なユーザ数	6	16
攻撃者数	1	4
サーバ数	3	2

たトポロジである．DFN トポロジは XC より複雑でより現実的なトポロジである．このトポロジはより現実に近い環境での効果を確認するのに適している．

シミュレーション評価は各トポロジから選択したルータのみで行う．表 2 のとおり，各トポロジにおけるルータ数は多く，そのすべての結果を記載することは難しい．そこで，上流にあるルータ，下流で攻撃者と隣接したルータ，下流で正常なユーザのみと隣接したルータの三種類の特徴的なルータを選択し，その結果を用いて評価を行った．

4.2　XC トポロジにおけるキャッシュヒット率

XC トポロジにおけるキャッシュヒット率の変化の結果を示す．XC トポロジでは，図 4 で示された 3 つのルータを選択して評価を行った．ルータ 5 はサーバと隣接した上流のルータであり，ルータ 8 は攻撃者と隣接したルータ，ルータ 6 は正常なユーザのみと隣接したルータである．

図 6 に Lightweight attack detection を用いた場合の XC トポロジにおけるキャッシュヒット率を示す．横軸は時間，縦軸はキャッシュヒット率を示し，60 分の時点から攻撃が開始される．図 6 より，ルータ 8 のキャッシュヒット率が攻撃開始と同時に大幅に低下していることがわかる．一方で，ルータ 5 とルータ 6 のキャッシュヒット率に大きな変化は見られなかった．このことから，攻撃者と隣接しているルータは攻撃の影響を大きく受けることがわかる．

図 7 に各ルータにおける攻撃検知の様子を示す．横軸は時間，縦軸は正規化された要求率変化量を示し，この値が 0 を超えると攻撃が検知される．ルータ 8 においては攻撃開始と同時に要求率変化量が 0 を超えており，攻撃が検知されている．ルータ 5 においては要求率変化量が上昇する

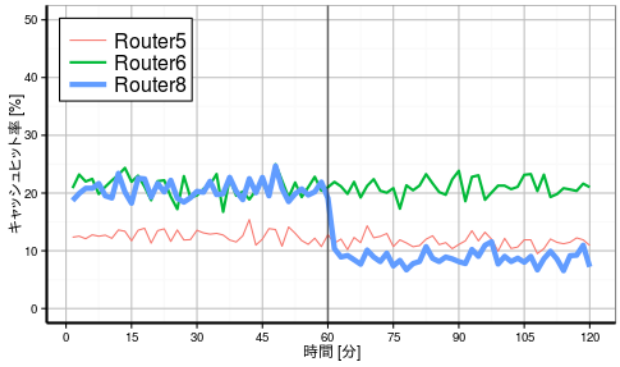

図 6　Lightweight attack detection を用いた XC トポロジにおけるキャッシュヒット率

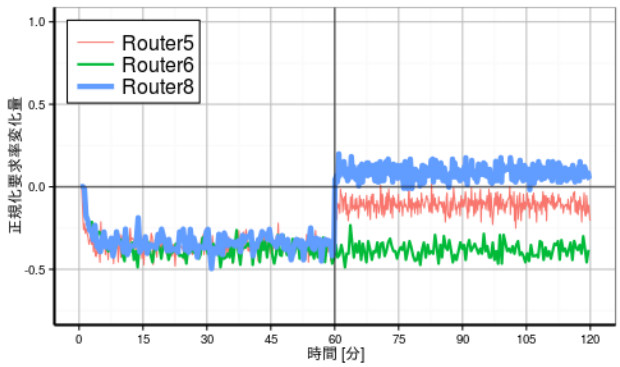

図 7　Lightweight attack detection を用いた XC トポロジにおける攻撃検知

ものの，その値は 0 付近でとどまっている．ルータ 6 の要求率変化量はほとんど見られなかった．攻撃検知の結果から，ルータ 8 では攻撃の影響が強く，ルータ 5 では攻撃の影響は見られるものの，それほど強くないことがわかる．ルータ 6 では攻撃の影響がほぼないことがわかる．キャッシュヒット率と攻撃検知の両方の結果から，ルータ 5 とルータ 6 では攻撃の影響が無視できるほど小さく，RMCP は攻撃の影響を強く受けているルータ 8 においてその影響を抑制する必要がある．

RMCP を用いた場合の XC トポロジにおける結果を示す．図 8 に RMCP を用いた場合のキャッシュヒット率を示す．この図より，ルータ 8 のキャッシュヒット率は攻撃前後でそれほど変化していないことがわかる．ルータ 5，ルータ 6 のキャッシュヒット率は，Lightweight attack detection の場合とあまり差がない．この結果から，RMCP はルータ 8 において Cache pollution attack によるキャッシュヒット率の低下を抑制できていると考えられる．次に，図 9 の RMCP を用いた場合の攻撃検知の結果を見てみる．攻撃開始と同時にルータ 8 とルータ 5 において要求率変化量の急上昇が見られるが，どちらもすぐに低下し，攻撃前の水準に戻っていることがわかる．攻撃検知の結果から，ルータ 5 とルータ 8 において攻撃が検知され，RMCP が

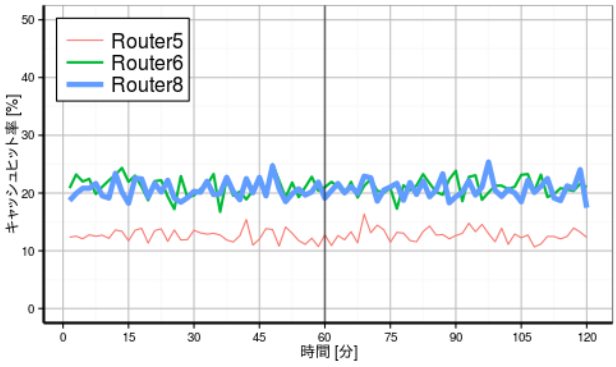

図 8　RMCP を用いた XC トポロジにおけるキャッシュヒット率

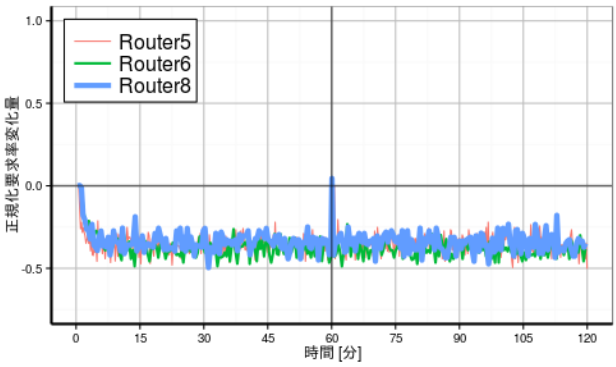

図 9　RMCP を用いた XC トポロジにおける攻撃検知

実行されたことがわかる．RMCP が実行されると，攻撃者が要求しているプレフィックスを階層性に基づいて識別し，ブラックリストを作成する．このブラックリストを用いて CS を攻撃前の状態に戻し，再び汚染されるのを防ぐことで要求率変化量を元の水準に戻し，キャッシュヒット率低下を抑制できたと考えられる．

4.3　DFN におけるキャッシュヒット率

現実的なトポロジにおける RMCP の効果を検証するために，DFN トポロジでも評価を行った．DFN トポロジで選択したルータは図 5 で示した通りである．上流のルータはルータ 2，下流で攻撃者と隣接したルータはルータ 5，下流で正常なユーザのみと隣接したルータはルータ 0 である．

図 10 に DFN トポロジのおいて Lightweight attack detection を用いた場合のキャッシュヒット率の変化を示す．攻撃開始と同時に，攻撃者と隣接したルータであるルータ 5 においてキャッシュヒット率の低下がみられた．一方で，XC トポロジの場合と同様に攻撃者から離れたルータはキャッシュヒット率の低下はほとんど見られなかった．XC トポロジの場合と異なる点として，上流のルータにおけるキャッシュヒット率が攻撃前の状態でも低い値をとっている．これは，DFN トポロジは XC トポロジより複雑なトポロジであり，上流と下流の距離がより離れていることが原因と思われる．DFN トポロジの場合においては，ルー

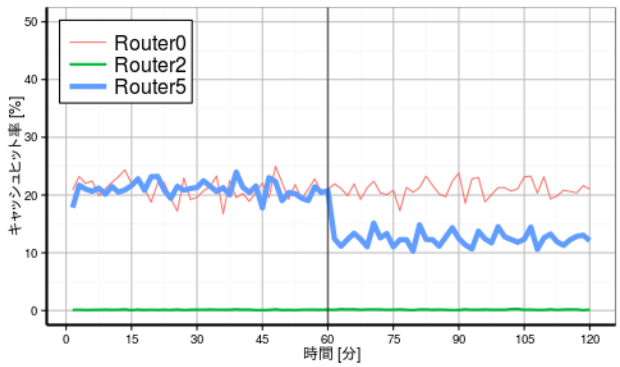

図 10 Lightweight attack detection を用いた DFN トポロジにおけるキャッシュヒット率

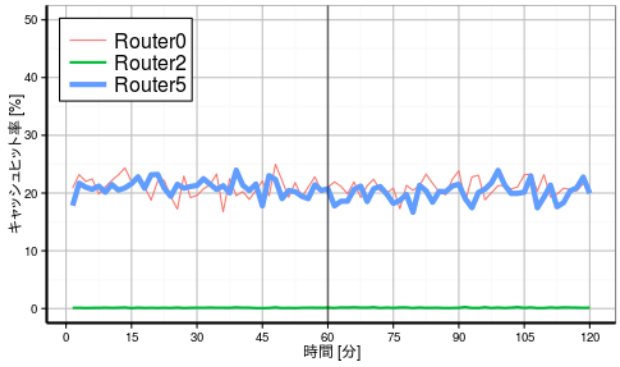

図 11 RMCP を用いた DFN トポロジにおけるキャッシュヒット率

タ 5 におけるキャッシュヒット率の低下を抑制する必要がある．

図 11 には RMCP を用いた場合のキャッシュヒット率の変化を示す．RMCP を用いた結果を見ると，ルータ 5 におけるキャッシュヒット率の低下が見られない．このことから，より現実的なトポロジにおいても RMCP は攻撃者の近くに位置し，攻撃の影響を強く受けるルータにおいてキャッシュヒット率低下を抑制できていることが確認された．Lightweight attack detection の考え方に基づいた攻撃検知，RMCP による攻撃の影響抑制は各ルータが独立に行う．よって，トポロジの影響を受けないためにより現実的なトポロジにおいても抑制効果が確認できたと考えられる．

5. 結論

各ルータが CS というキャッシュを持つ NDN において，Cache pollution attack というキャッシュに対する攻撃が問題とされている．この問題に対して，関連研究の多くはコンテンツ名に着目し，Cache pollution attack の一種である Locality-disruption 攻撃に焦点を当てている．そこで，本稿ではコンテンツ名のプレフィックスが持つ階層性に着目し，False-locality 攻撃の影響を抑制する手法である RMCP を提案した．この手法はプレフィックスの階層性に基づいて攻撃者が要求するプレフィックスを識別し，その情報を用いることでルータにおける正常なユーザからの要求のキャッシュヒット率低下を抑制するものである．

シミュレーションの結果から，RMCP は攻撃の影響を強く受けるルータにおいてキャッシュヒット率の低下を抑制することを確認した．また，2 つのトポロジの両方で攻撃の抑制効果を確認し，トポロジの影響を受けにくいことがわかった．以上のことから，RMCP によるプレフィックスの階層性に基づいた手法は Cache pollution attack の影響抑制に効果的であることを確認した．

参考文献

[1] Jacobson, V., Smetters, D. K., Thornton, J. D., Plass, M. F., Briggs, N. H. and Braynard, R. L.: Networking Named Content, *Proceedings of the 5th International Conference on Emerging Networking Experiments and Technologies*, CoNEXT '09, pp. 1–12 (2009).
[2] Zhang, L., Afanasyev, A., Burke, J., Jacobson, V., claffy, k., Crowley, P., Papadopoulos, C., Wang, L. and Zhang, B.: Named Data Networking, *ACM SIGCOMM Computer Communication Review (CCR)*, Vol. 44, No. 3, pp. 66–73 (2014).
[3] Deng, L., Gao, Y., Chen, Y. and Kuzmanovic, A.: Pollution Attacks and Defenses for Internet Caching Systems, *Comput. Netw.*, Vol. 52, No. 5, pp. 935–956 (2008).
[4] Conti, M., Gasti, P. and Teoli, M.: A lightweight mechanism for detection of cache pollution attacks in Named Data Networking, *Computer Networks*, Vol. 57, No. 16, pp. 3178–3191 (2013).
[5] Ali, W., Shamsuddin, S. M. and Ismail, A. S.: A survey of web caching and prefetching, *International Journal of Advances in Soft Computing and its Application*, pp. 18–44 (2011).
[6] Balamash, A. and Krunz, M.: An overview of web caching replacement algorithms, *Communications Surveys Tutorials, IEEE*, Vol. 6, No. 2, pp. 44–56 (2004).
[7] Stefan Podlipnig, L. B.: A survey of Web cache replacement strategies, *ACM Computing Surveys*, Vol. 35, No. 4, pp. 374–398 (2003).
[8] Manivel, V., Ahamad, M. and Venkateswaran, H.: Attack Resistant Cache Replacement for Survivable Services, *Proceedings of the 2003 ACM Workshop on Survivable and Self-regenerative Systems: In Association with 10th ACM Conference on Computer and Communications Security*, SSRS '03, pp. 64–71 (2003).
[9] Park, H., Widjaja, I. and Lee, H.: Detection of cache pollution attacks using randomness checks, *IEEE International Conference on Communications (ICC)*, pp. 1096–1100 (2012).
[10] Xie, M., Widjaja, I. and Wang, H.: Enhancing cache robustness for content-centric networking, *INFOCOM, 2012 Proceedings IEEE*, pp. 2426–2434 (2012).
[11] NS-3 Consortium: ns3, `http://www.nsnam.org/` (2015).
[12] Afanasyev, A., Moiseenko, I. and Zhang, L.: ndnSIM: NDN simulator for NS-3, Technical report, NDN (2012).
[13] Heckmann, O., Piringer, M., Schmitt, J. and Steinmetz, R.: On Realistic Network Topologies for Simulation, *Proceedings of the ACM SIGCOMM Workshop on Models, Methods and Tools for Reproducible Network Research*, MoMeTools '03, pp. 28–32 (2003).