شبکه سایه و کاربری آن
در مدیریت و راهبری استراتژیک پدافند دفاعی و جنگ سایبری:
با نگاهی به تحولات سیاسی منطقه خاورمیانه و موج بیداری اسلامی

نوید کمالی

دانشجوی مهندسی نرم افزار ، دانشگاه آزاد اسلامی واحد نیشابور

Info@nKamali.ir

چکیده

با رشد و گسترش دانش در سطح جهان و بین تمدن‌های مختلف بشری ما وارد عصر جدیدی از شیوه‌های نوین ارتباطات شده‌ایم. امروزه نسل جدید ارتباطات رادیوی و اینترنتی تمام جنبه‌های زندگی بشر هزاره جدید را تحت تاثیر قرار داده‌اند. متکی شدن تمدن‌ها به شیوه‌های ارتباطی خاص و استاندارد جهانی باعث شده تا غالبا کاربران عمومی این سیستم‌ها مورد سوء استفاده سرویس دهندگان یا کنترل کنندگان منطقه ای یا جهانی این خدمات قرار گیرند. ارتباطات مخابراتی و اینترنتی که با هدف گسترش حقوق بشر و کمک به رشد تمدنها طراحی شده بودند در سالهای اخیر به ابزاری برای به انزوا کشاندن و کنترل و شنود کاربران در سطح جهانی تغییر کاربری داده اند. از این رو طراحی نسل جدیدی از ارتباطات رادیویی و اینترنتی مستقل و بدون نیاز به سرویس دهنده (Server Less)مبتنی بر تکنولوژی شبکه‌های مش (Mesh Network) و شبکه‌های نظیر به نظیر (P2P)می تواند راهکاری برای کمک به رشد تمدنها و گسترش حقوق بشر و همچنین به ابزاری برای گسترش ارتباطات در حین بحرانهای کشوری و جهانی (موج بیداری اسلامی در خاورمیانه و شمال آفریقا ,...) که استفاده از شیوه‌های رایج با مشکلات یا ریسکهای امنیتی همراه است(عملیاتهای نظامی و اطلاعاتی , مناطق جنگی یا بحران زده و...) بهره برد. در این مقاله تلاش خواهم نمود تا به تشریح این نوع شبکه که مبنای پروژهتحقیقاتی - عملیاتی خود در این حوزه با نام ابابیل است بپردازم.

کلمات کلیدی

شبکه سایه ,[1] دفاع سایبری,حقوق بشر , امنیت اطلاعات , اینترنت خصوصی , آزادی اطلاعات

۱- مقدمه

با وقوع بحرانهای سیاسی و اجتماعی اخیر در سطح جهان و خصوصا منطقه خاورمیانه و شمال آفریقا همیشه مقوله ارتباطات و خصوصا نوع خاص آن که ارتباطات مخابراتی و اینترنت است به عنوان ابزازی برای کنترل و سرکوب مخالفان توسط رژیمهای دیکتاتوری تبدیل شده است البته از سالها پیش طراحی و ساخت نسل از جدیدی از ارتباطات برای نیروهای حاظر در میادین جنگی مانند سپاه پاسداران انقلاب اسلامی ایران و ارتش جمهوری اسلامی ایران و یا گروههای مقاومت و سرویسهای اطلاعاتی در کشورهای تحت اشغال نیروهای اشغالگر مانند گروههای حزب الله لبنان و جنبش آزادی بخش فلسطین مد نظر بنده و بسیاری از کارشناسان و محققان این عرصه در سطح جهان بوده و می‌باشد.

ارتباطات رایج فعلی در سطح جهان مبتنی بر سرویس دهنده و سرویس گیرنده می‌باشد و تا زمانی سرویس برقرار می‌باشد که سرویس دهنده سرویس خود را برای گیرندگان آن سرویس ارائه دهد و در صورت عدم ارائه سرویس از جانب سرویس دهنده دیگر سرویس گیرندگان به سرویس دسترسی نخواهند داشت. [۲]

در حالت یا سناریوی دیگر اگر سرویس دهنده توسط نیروهای غیر قابل اعتماد کنترل شود تمام ارتباطات مبتنی بر سرویس آن سرویس دهنده برای سرویس گیرندگان شنود و رصد خواهد شد و احتمال ایجاد مخاطرات امنیتی فراوان برای کاربران وجود خواهد داشت.

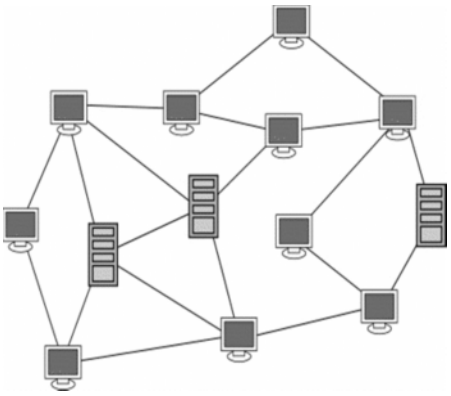
گاهی هم شرایط بگونه ای می‌یاشد که امکان استفاده از شیوه‌های رایج ارتباطی مبتی بر سرویس دهنده و سرویس گیرنده ممکن نیست. بطور مثال در یک منطقه بحران زده که تمام زیرساخت‌های ارتباطی و مخابراتی آن توسط عواملی از جمله طوفان, سیل , بمباران و یا بسیاری از عوامل طبیعی و غیر طبیعی از بین رفته اند در این صورت نیاز است تا با کمترین هزینه و در کمترین زمان ممکن و با بیشترین میزان امنیت و اعتماد شبکه‌های مخابراتی و ارتباطی را برای کاربری بمنظور ایجاد کانالهای ارتباطی ویژه کنترل و مدیریت بحران و هماهنگ سازی نیروهای امدادی و امنیتی مورد نیاز بکار گرفت.

برای این منظور مدلهای مختلفی را بمنظور طراحی بستر این نوع شبکه مورد بررسی قرار داده و هریک را از ابعاد مختلف مورد نقد و بررسی قرار دادیم.

با توجه به نیازهای پروژه که شامل ۱- مستقل بودن ۲- انعطاف پذیر بودن در شرایط مختلف ۳- ارزان و در دسترس بودن تجهیزات ۴- امنیت بالا ۵- کاربر محور بودن ۶-قابلیت گسترش و آسان ۷- قابلیت ارائه خدمات متنوع ارتباطی

پس از مطالعات و بررسی‌های فراوان شبکه‌های و توپولوژی‌های مختلف , توپولوژی مش را به عنوان بستری مناسب برای پیاده سازی این نوع خاص از شبکه‌های مخابراتی برگزیده شد.

ارتباط اعضای شبکه در برخی از قسمت در شبکه مش توسط کابل بوده ، برخی دیگر توسط فیبر نوری در برخی از مسیرها نیز امواج ماکروویو و سیستم‌های ماهواره ای استفاده می‌شود. [۳]



شکل (۱): همبندی مش

همانطور که در شکل(۱) ملاحظه می‌کنید، در این توپولوژی هر عضو شبکه یک سیستم مستقل می‌باشد و برای رسیدن به مقصد مسیرهای متعددی در پیش روی دارد که این خود باعث پایداری بالای شبکه در شرایط دشوار می‌شود.

اصل طراحی توپولوژی مش هم بر پایدار نگه داشتن شبکه در صورت قطع دسترسی و یا از کار افتادن هر یک از گره‌های ارتباطی شبکه پایدار است.

نوعی دیگری از شبکه‌ها که با کاربریهای مورد نظر ما همخوانی دارد شبکه‌های نظیر به نظیر (Peer to Peer)است.

در این نوع از ارتباط ایستگاه کاری ویژه ای جهت نگهداری بانکهای اطلاعاتی و داده‌ها وجود ندارد که این مهمترین مزیت این نوع

ارتباط در کاربرهای اطلاعاتی است که امنیت ارتباط در آنها از اهمیت بالا و حیاتی برخوردار است.

طراحی شبکه ای که بتواند از ویژگی‌های این دو نوع شبکه فوق الذکر بهره ببرد می‌تواند گامی بزرگ در عرصه‌های رقابتی مانند جنگهای اطلاعاتی که در آن نیاز به راه اندازی شبکه‌های ارتباطی ویژه جهت سرویس‌های جاسوس و گروه‌های آزادی بخش و مقاومت است, باشد.

از جمله فواید این نوع شبکه تلفیقی می‌توان به موارد زیر اشاره کرد:

- مخفی بودن فرستنده و گیرنده اطلاعات برای مهاجمین
- ایجاد یک رسانه قابل اعتماد جهت انتقال اطلاعات مستقل از شبکه‌های موجود مخابراتی تحت کنترل دشمن
- تضمین امنیت اطلاعات از مبدا تا مقصد بوسیله رمزنگاری ترافیک
- تکذیب پذیر بودن اطلاعات ذخیره شده برای کاربر
- غیرقابل مسدود کردن و ازکار انداختن شبکه توسط مهاجمین
- ذخیره‌سازی و مسیریابی پویای اطلاعات
- تمرکززدایی از تمام سرویس‌ها و توابع شبکه
- در دسترس و ارزان بودن تجهیزات سخت افزاری مورد نیاز برای عموم
- سازگاری با انواع شیوه‌های ارتباطی (متن,صوت,تصویر)

۲- مدل مسیریابی

شبکه سایه یک شبکه پویا نظیر به نظیر تشکیل شده از تعداد زیادی از گره‌ها است که اطلاعات را از یکدیگر دریافت یا برای یکدیگر با استفاده از زیر ساختهای یک شبکه رسمی بصورت مخفی ارسال می‌کنند.

شبکه نظیر به نظیر بدلیل دارا بودن سرعت و نرخ بالای انتقال اطلاعات از سالها پیش بعنوان یک بستر مناسب جهت به اشتراک گذاری اطلاعات پر حجم در شبکه بکار برده می‌شود. ولی این نوع شبکه به تنهایی قابلیت مخفی سازی هویت کاربران و رمز نگاری اطلاعات را ندارد.

شیوه‌های مختلفی برای پیاده سازی شبکه سایه می‌توان ارائه داد که هریک بنا بر نوع کاربری خاص می‌توانند برای موقعیت‌های مختلف مورد استفاده قرار بگیرند.

یک مدل کاربردی را اینگونه می‌توان تشریح کرد ; شبکه سایه بر مبنا از تعداد زیادی گره N و مقدار زیاد از اطلاعات D تشکیل شده است که با فرض قرار گرفتن x کاربر پشت هر گره که دارای زیر مجموعه اطلاعات D_x از کل طلاعات موجود در شبکه است.

هر کاربر x مایل است که اطلاعات را تنها با گره هایی که به آنها اعتماد کامل دارد به اشتراک بگذارد که ما آنها را F_x می‌نامیم.

با فرض اینکه رابطه دوستی دارای خاصیت جابجایی باشد برای هر دو گره x و ,y اگر x در گروه F_y باشد پس y در گروه F_x خواهد بود.

هر آیتم داده d دارای یک مشخصه A_d به همراه خود است. هر مشخصه دارای تعدادی جفت داده‌های عددی است که هر جفت آن یک مشخصه داده را مشخص می‌سازد.

هر گره x دارای یک ارتباط رمزنگاری شده امن با گره‌های دیگر موجود در F_x است. نکته مهم عدم وجود هیچ زیرساخت مرکزی ثابت برای تصدیق هویت است که باعث افزایش امنیت تبادل اطلاعات در طول مسیر می‌شود زیرا تمام فرایندهای رمزنگاری و رمز گشایی درگره‌ها صورت می‌گیرد و هر گره دارای کلیدهای رمز گذاری و رمز گشایی منحصر به فرد خود است و از این رو هر گره می‌تواند کلید عمومی خود را با گره‌های قابل اعتماد زیر مجموعه خود به اشتراک بگذارد.

حال بهتر است به تشریح الگویی متفاوت از الگوی بالا که آن را T2T یا (Trust to Trust) می‌نامیدم بپردازیم.

اگر شبکه ما داری U کاربر و R مسیریاب باشد پس شبکه N ما را می‌توان این گونه تعریف کرد $N = U \cup R$ پس این حالت نیز صادق خواهد بود $R \cap U \neq 0$.

هر گره شامل U و R می‌شود که این یعنی علاوه بر کاربر مستقل بودن یک مسیریاب هم برای کل شبکه سایه محسوب می‌شود.

با فرض اینکه کلیه گره‌های ما با یکدیگر یک مدار را تشکیل دهند (حداقل طول مدار پیشفرض در این مدل l=3) ; هر جفت مسیریاب و کاربر (U+R) مجموعه ای از کلیدهای مخفی را به اشتراک می‌گذارد هرچند مسیریاب از هویت کاربرانی که کلیدها متعلق به آنهاست اطلاعی ندارد و فرض ما در اینجا این است که سیستم توزیع و به اشتراک گذاری کلیدها مستقل از پروتکل ارتباطی ماست. اگر K را فضای کلید در نظر بگیریم سه گانه (u,r,i) به کلید iیمکه توسط کاربر u و مسیریاب r به اشتراک گذاشته شده است اشاره می‌کند.

حال اگر P نمادی برای پیامهای کنترل باشد پس $\overline{P}$ بسطی برای P در شرایطی خواهد بود که بوسیله l کلید رمزنگاری شده باشد.

هر پیام کنترل بوسیله یک مشخص کننده لینک و مدار در هنگام ارسال تگ می‌شود که این خود باعث می‌شود تا پیام مربوط به پروتکل ارتباطی ما به صورت $M = N_+ \times N_+ \times P$ شود.

ما برای رمزنگاری پیغام $p \in P$ بوسیله کلید k از $\{p\}_k$ و برای رمز گشایی از $\{p\}_{-k}$ استفاده می‌کنیم. پس یک پیغام که توسط یک کلید رمز دیگر رمز دوباره شده است بصورت $\{\{p\}_{k1}\}_{k2}$ خواهد بود که در اینجا به اختصار آن را به این صورت نمایش می‌دهم $\{p\}_{k1,k2}$.

برای ایجاد یک ارتباط u پیغام $\{CREATE\}_{k1}$ را به اولین مسیریاب (r_1) مدار ارسال می‌کند. در ادامه پیغام رمزنگاری شده با کلید k_1 بین u و r_1 به اشتراک گذاشته می‌شود.

در اینجا r_1 بوسیله تکرارهای پی در پی برای رمزگشایی پیام بوسیله کلیدهایی در دسترس خود k_1 را شناسایی می‌کند و پیغام CREATED را ارسال می‌کند.

با توجه به مدار ,u مسیریاب r_iدیگری را به انتهای مدار با ارسال پیغام $\{[EXTEND, r_i, \{CREATE\}_{ki}]\}_{ki-1,...,k1}$

هر چه پیام در مدار به انتهای مسیر نزدیکتر می‌شود , مسیریابها پیغام موجود در آن را به نوبت رمزگشایی می‌کنند و همانطور که در بالا نشان دادم R_{i-1} مرحله CREATE را انجام می‌دهد و پیغام $\{EXTENDED\}_{ki-1}$را بر می‌گرداند.

پروسه کامل را می‌توان در شکل (۲) مشاهده نمایید.

```
1: c ∈ {(r_1,...,r_l) ∈ R^l | ∀_i r_i ≠ r_{i+1}}; init: arbitrary        ▷ User's circuit
2: i ∈ N; init: random                                                  ▷ Circuit identifier
3: b ∈ N; init: 0                                                       ▷ Next hop to build
4: procedure START
5:     SEND(c_1, [i, 0, {CREATE}_{k(u,c,1)}])
6:     b = 1
7: end procedure
8: procedure MESSAGE(msg,j)                                  ▷ msg ∈ M received from j ∈ N
9:     if j = c_1 then
10:        if b = 1 then
11:            if msg = [i, 0, CREATED] then
12:                b++
13:                SEND(c_1, [i, 0, {[EXTEND, c_b, {CREATE}_{k(u,c,b)}]}_{k(u,c,b-1),...,k(u,c,1)}])
14:            end if
15:        else if b < l then
16:            if msg = [i, 0, {EXTENDED}_{k(u,c,b-1),...,k(u,c,1)}] then
17:                b++
18:                SEND(c_1, [i, 0, {[EXTEND, c_b, {CREATE}_{k(u,c,b)}]}_{k(u,c,b-1),...,k(u,c,1)}])
19:            end if
20:        else if b = l then
21:            if msg = [i, 0, {EXTENDED}_{k(u,c,b-1),...,k(u,c,1)}] then
22:                b++
23:            end if
24:        end if
25:    end if
26: end procedure
```

شکل (۲): فرآیند کار ماشین

۲-۱- کلیدها و سیستم جستجو

در شبکه سایه هویت دهی بسته‌های اطلاعاتی به اشتراک گذاشته شده از طریق قرار دادن یک رشته درهم سازی (هش) شده در هر بسته اطلاعاتی انجام می‌گیرد.

با استفاده از استاندارد پیشفرض 160-bit SHA-1[4] یا بالاتر می‌توان امنیت تبادل اطلاعات را تا حد بالایی تضمین کرد.

ساده ترین نوع کلید فایل Keyword-signed key (KSK) است که از رشته ی کوتاهی از متن توصیفی ای به دست می‌آید که کاربر در هنگام ارسال در شبکه انتخاب کرده است.

برای مثال کاربری که مقاله‌ای درباره جنگ text/philosophy/sun-tzu/art-of-war را وارد می‌کند ممکن است از این توصیف برای آن استفاده کند. از این رشته به عنوان داده برای تولید قطعی یک جفت کلید عمومی /خصوصی استفاده می‌شود. نیمه ی عمومی آن سپس برای به دست آوردن کلید فایل درهم سازی می‌شود.

از قسمت خصوصی جفت کلید نامتقارن برای علامت گذاری فایل استفاده می‌شود، تا بررسی کوچکی پدید آید که یک فایل بازیابی شده با کلید فایل مطابقت داشته باشد. اگرچه باید توجه داشت که یک مهاجم می‌تواند با گرد آوردن یک لیست از رشته‌های توصیفی از یک dictionary attack برعلیه این نشانه کذاری (امضا) استفاده کند. فایل همچنین با استفاده از خود رشته ی توصیفی به عنوان یک کلید، رمزدار می‌شود

برای دادن اجازه ی بازیابی داده به دیگران، کافیست کاربر رشته ی توصیفی را منتشر کند. این کار به یاد آوردن کلیدهای-keyword signed را راحت و انتقال آن به دیگران را آسان می‌کند. اما این رشته‌ها سیستم نامگذاری جهانی بی تنوعی را به وجود می‌آورند که مشکل آفرین است. برای مثال هیچ چیز مانع از این نیست که دو کاربر مستقلا رشته ی توصیفی یکسانی را برای فایل‌های متفاوت انتخاب کنند یا فایل‌های هرز را با نام‌های محبوب وارد شبکه کنند.

این مشکلات به وسیله ی signed-subspace key (SSK) که سیستم نامگذاری شخصی را امکان پذیر می‌سازد مورد توجه قرار گرفتند. یک کاربر به وسیله ی ساختن جفت کلید عمومی/خصوصی که به عنوان شناسه ی سیستم (فضا) نامگذاریش عمل میکند، سیستم نامگذاری شخصی خود را به وجود می‌آورد. برای وارد کردن یک داده، کاربر همچون گذشته یک رشته متن توصیفی انتخاب می‌کند. کلید عمومی سیستم نامگذاری و رشته ی توصیفی به صورت جدا گانه هش (درهم سازی) می‌شوند، با همXOR می‌شوند و سپس دوباره برای به دست آمدن کلید داده هش می‌شوند.

در keyword-signed key از بخش خصوصی جفت کلید نامتقارن برای علامت گذاری داده استفاده می‌شود. این علامت (امضا)، که از یک جفت کلید تصادفی به دست می‌آید، امن تر از علایمی است که برای keyword-signed keys به کار می‌رود. داده، همچنین، همچون گذشته به وسیله ی رشته ی توصیفی رمزگذاری می‌شود.

برای دادن اجازه ی بازیابی فایل به دیگران، کاربر رشته ی توصیفی و کلید عمومی فضای فرعی خودش را منتشر می‌کند. ذخیره کردن اطلاعات به کلید خصوصی احتیاج دارد بنابراین مالک فضای فرعی (subspace) تنها کسی است که می‌تواند به آن داده اضافه کند.

اکنون مالک توانایی مدیریت سیستم نامگذاری خود را دارد. برای مثال او می‌تواند به وسیله ی ساخت فایل‌های راهنما-مانند حاوی هایپرتکست‌های اشاره گر به فایل‌های دیگر، یک ساختار سلسله مراتبی را شبیه سازی کند. در صورتی که از ترکیب زبانی مناسبی استفاده شود که برای کاربر (سرویس گیرنده) قابل درک باشد. راهنماها همچنین می‌توانند اشاره گرهای دیگر را نشان بدهند.

سومین نوع کلید content-hash key (CHK) است که برای انجام آپدیت و جداسازی مفید واقع می‌شود. یک content-hash key تنها از هش کردن مستقیم محتوای فایل مربوطه به دست می‌آید. این کار به هر فایل یک کلید فایل شبه منحصر به فرد می‌دهد. داده‌ها همچنین به وسیله ی یک کلید رمزگذاری تصادفی رمزنگاری می‌شوند. برای دادن اجازه ی بازیابی داده به دیگران، کاربر خود content-hash key را به همراه کلید آشکارسازی منتشر می‌کند. توجه داشته باشید که کلید آشکارسازی هرگز با فایل ذخیره نمی شود؛ بلکه تنها به همراه کلید داده منتشر می‌شود.

Content-hash keyها زمانی که به همراه signed-subspace keyها و با استفاده از یک مکانیزم غیرمستقیم به کار گرفته شوند بیشترین کارایی را دارند. برای ذخیره کردن یک فایل قابل بروز رسانی، کاربر ابتدا آن را تحت content-hash key وارد می‌کند و سپس یک فایل غیرمستقیم را تحت یک signed-subspace key وارد می‌کند که محتوایش همان content-hash key است. این کار دیگران را قادر می‌سازد تا فایل را با داشتن signed-subspace key در دو مرحله بازیابی کنند.

برای بروز رسانی یک داده یا فایل، دارنده ابتدا یک نسخه جدید را تحت content-hash key آن که باید با content-hashنسخه قبلی تفاوت داشته باشد وارد می‌کند. او سپس یک داده یا فایل غیرمستقیم جدید را تحت signed-subspace keyاصلی وارد می‌کند که نسخه بروز رسانی شده را نشان می‌دهد. زمانی که این وارد کردن به یک گره می‌رسد که دارای نسخه قدیمی است، یک تلاقی کلیدی اتفاق می‌افتد. گره ، نشانه(امضا) ی روی نسخه جدید را چک و معتبربودن و جدیدتر بودن نسبت به نسخه فعلی آن را تایید کرده و نسخه جدید را جایگزین نسخه قدیمی می‌کند. بدین ترتیب، signed-subspace key کاربر را به جدیدترین نسخهداده یا فایل هدایت می‌کند، در حالی که نسخه‌های قدیمی همچنان به وسیله ی content-hash key مستقیماً قابل دسترسی خواهند بود. (گرچه، اگر درخواستی برای این داده‌ها وجود نداشته باشد، این نسخه‌های قدیمی به تدریج از شبکه حذف خواهند شد). از این مکانیزم می‌توان هم در راهنماها و هم داده‌های عادی استفاده کرد.

۲-۲- مبارزه با آنالیز ترافیک شبکه

از content-hash keyها می‌توان برای تقسیم کردن فایل‌ها به چندین قسمت استفاده کرد. برای فایل‌های بزرگ، تقسیم کردن، به دلیل محدودیت‌های ذخیره سازی و پهنای باند می‌تواند خوشایند باشد. حتی تقسیم کردن فایل هایی با اندازه ی متوسط به بخش هایی با اندازه ی استاندارد (برای مثال 2^n کیلوبایت) دارای فوایدی در مبارزه با آنالیز ترافیک است. این کار به راحتی و با وارد کردن هر قسمت به صورت جداگانه تحت یک content-hash key و ساختن یک فایل غیرمستقیم (یا چندین سطح از فایل‌های غیرمستقیم) برای نشان دادن هر قسمت امکان پذیر می‌باشد. [۵]

با وجود همه ی این ها، مشکل پیدا کردن کلیدها همچنان پابرجا باقی می‌ماند. آسان ترین راه برای اضافه کردن قابلیت جستجو به فری نت اجرای یک hypertext spider مشابه آنهایی است که برای جستجو در وب مورد استفاده قرار می‌گیرند. گرچه این راه حل ممکن است از بسیاری جهات راه حل جذابی به نظر برسد، اما با هدف اجتناب از متمرکزسازی منافات دارد. راه حل امکان پذیر دیگر، ساخت یک دسته ی به خصوص از فایل‌های غیرمستقیم کم حجم است. هنگامی که یک فایل اصلی وارد می‌شود، کاربر می‌تواند تعدادی فایل غیرمستقیم که هر یک شامل یک اشاره گر به فایل اصلی هستند و با توجه به کلیدواژه ی جستجو نامگذاری شده اند را به همراه فایل اصلی وارد کند. این فایل‌های غیرمستقیم از این جهت که می‌توان چندین فایل با کلید یکسان (برای مثال کلید واژه ی جستجو) را کنار هم داشت، با فایل‌های عادی تفاوت دارند. درخواست برای این کلیدها به جای آنکه در اولین فایل پیداشده متوقف شود، تا زمانی که تعداد معینی از نتایج روی هم انبار شوند به جستجو ادامه خواهد داد. مدیریت این حجم بالا از فایل‌های غیر مستقیم یک مشکل حل نشده است.

یک مکانیزم جایگزین تشویق افراد به ساخت مجموعه هایی از کلیدهای مورد علاقه و عمومی سازی کلیدهای این مجموعه هاست. از این روش به صورت گسترده در شبکه ی ارتباطی جهانی استفاده می‌شود.

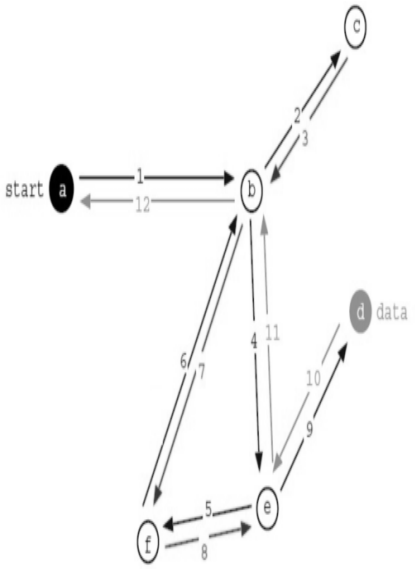
۲-۳- بازیابی اطلاعات

برای بازیابی یک فایل، کاربر ابتدا باید کلید فایل دوتایی (باینری) آن را به دست بیاورد یا محاسبه کند. او سپس پیام درخواستی به گره خود می‌فرستد و آن کلید و مقدار زنده ماندن پیام در سطح شبکه (hops-to-live value)را مشخص میکند. وقتی که یک گره درخواستی را دریافت می‌کند، ذخیره ی خود را برای آن داده جستجو می‌کند و در صورت یافتن آن، آن را به همراه یک یادداشت که گره را منبع داده معرفی می‌کند، به کاربر پس می‌دهد. در صورت یافت نشدن، گره نزدیک ترین کلید به کلید درخواست شده را در جدول مسیریابی خود پیدا می‌کند و درخواست را به گره مربوطه می‌فرستد. اگر درخواست درنهایت موفقیت آمیز باشد و با داده ی درخواستی بازگردد، گره داده را به کاربر درخواست دهنده می‌فرستد، فایل را در ذخیره ی داده‌های خود ذخیره می‌کند و یک ورودی جدید مربوط به منبع اصلی داده و کلید درخواست شده در جدول آدرس دهی خود می‌سازد.

درخواست‌های بعدی برای همان کلید به سرعت و با استفاده از اطلاعات ذخیره شده پاسخ داده می‌شود. درخواست برای یک کلید مشابه (تشابه به وسیله ی مشابهت واژگانی مشخص می‌شود) به منبع داده ای ارسال می‌شود که در درخواست قبلی موفق عمل نموده است. از آنجا که نگهداری یک لیست از منابع داده‌ها یک نگرانی امنیتی بالقوه است، هر گره می‌تواند در طول مسیر، به صورت یک طرفه تصمیم به تغییر پیام پاسخ بگیرد و خود یا یک گره دلخواه دیگر را به عنوان منبع داده معرفی کند.

اگر یک گره نتواند درخواست را به گره انتخاب شده بفرستد (گره هدف با مشکل مواجه است یا ارسال این درخواست یک حلقه ایجادمی کند)، گرهی که دومین کلید مشابه را دارد امتحان می‌شود، سپس سومین گرهی که کلید مشابه را داراست و الی آخر. اگر یک گره گزینه ی دیگری برای امتحان نداشته باشد، گزارش عدم موفقیتی به گره مجاور ارسال می‌کند و آن گره نیز دومین انتخاب خود را می‌آزماید.

اگر از حد hops-to-live فراتر رفته شود، یک پیام عدم موفقیت به درخواست دهنده ی اصلی فرستاده می‌شود، بدون اینکه گره دیگری امتحان شود. گره‌ها ممکن است به صورت خودسرانه hops-to-live valueهای بزرگ را برای کم شدن بار شبکه کوتاه کنند. آنها همچنین ممکن است پس از مدتی درخواست‌های درحال انتظار را برای خالی نگه داشتن حافظه ی پیام، به فراموشی بسپارند.



شکل (۳): توالی پیامهای درخواست

شکل (۳) یک توالی معمولی پیام‌های درخواست را نشان می‌دهد. کاربر درخواستی را در گره a وارد می‌کند. گره a درخواست را به گره b می‌فرستد، گره b نیز آن را به گره c ارسال می‌کند. گره c قادر به برقرای ارتباط با هیچیک از گره‌ها نیست و یک پیام "درخواست ناموفق" را به گره b ارسال می‌کند. گره b سپس گزینه ی دوم خود، e، را امتحان می‌کند. e نیز درخواست را به f می‌فرستد، گره f درخواست را به b ارسال می‌کند، b حلقه را تشخیص می‌دهد و یک پیام عدم موفقیت به عقب می‌فرستد. گره f قادر به برقراری ارتباط با هیچیک از گره‌های دیگر نیست و بنابراین پیام عدم موفقیتی را به یک قدم عقب تر، گره e ارسال می‌کند. گره e درخواست را به انتخاب دوم خود، d، می‌فرستد که داده را در اختیار دارد. داده از d و با عبور از e و b به a بازگردانده و از آنجا به کاربر ارسال می‌شود. داده همچنین در e، b و a نیز ذخیره می‌شود.

این مکانیزم چندین اثر دارد. مهم ترین تاثیر آن، این است که ما فرض می‌کنیم که کیفیت routing با گذشت زمان به دو دلیل افزایش می‌یابد: اول اینکه گره‌ها باید با گذشت زمان در پیدا کردن مجموعه کلیدهای مشابه تخصص پیدا کنند. اگر یک گره در routing tableها تحت یک کلید خاص لیست شده باشد، آن گره معمولا درخواست هایی برای کلیدهای مشابه همان کلید دریافت می‌کند. بنابراین، این احتمال وجود دارد که این گره "تجربه" ی بیشتری برای پاسخگویی به این گونه درخواست‌ها کسب کند و اطلاعات routing table هایش در این باره که کدام گره‌ها این کلیدها را حمل می‌کنند بیشتر شود. دوم اینکه، گره‌ها باید به صورت مشابهی در ذخیره کردن دسته فایل هایی با کلید مشابه نیز تخصص کسب کنند، به این علت که فرستادن موفقیت آمیز یک درخواست در نهایت یک کپی از فایل درخواست شده را نصیب گره می‌کند و چون اکثر درخواست ها، درخواست برای کلیدهای مشابه خواهند بود، گره معمولا فایل‌های مشابهی را به دست خواهد آورد. در مجموع، این دو اثر باید کارایی درخواست‌های آینده را در یک چرخه ی تقویتی افزایش دهند زیرا گره‌ها routing tableها و datastore هایی می‌سازند که بر مجموعه ی خاصی از داده‌ها تمرکز دارند و این کلیدها دقیقا همان هایی هستند که بیشترین درخواست را خواهند داشت. [۶]

۳- ویژگی‌های پروتکل

پروتکل مورد استفاده در شبکه سایه کاملا داده محور بوده و هر بسته اطلاعاتی قابلیت استفاده از دو تقسیم بندی رایج پورتهای ارتباطی یعنی TCP و UDP درا است.

هر بسته ارتباطی در طول تبادل در شبکه بین هر جفت گره یک شناسه خاص را دریافت کرده که این شناسه شیوه تبادل و نوع داده را برای گره دریافت کننده مشخص می‌سازد.

آدرس مقصد مقصد نهایی هر بسته اطلاعاتی بصورت کد شده در شناسه هر بسته وجود داشته ولی هر گره تنها را رسیدن به یک از گره‌ها را می‌داند. [۷]

هر ارتباط خاص در این نوع شبکه بوسیله یک درخواست آغاز می‌شود یعنی وقتی یک نود که غالبا سیستم کلاینت یا کاربر شبکه است از شبکه درخواست دریافت یا ارسال اطلاعات می‌کند. اگر سیستم مقابل پاسخ دهد ارتباط در شبکه آغاز خواهد شد و در غیر این صورت شبکه پاسخ منفی را برای کاربر ارسال خواه کرد.

پوشش عملیاتی پورتهای مختلف جهت تبادل اطلاعات علاوه بر ایجاد قابلیت استتارترافیک باعث جلوگیری از از کار افتادن شبکه توسط عوامل بیرونی مهاجم می‌گردد.

۳-۱- دنیای کوچک سایه

شبکه سایه را می‌توان بدلیل استفاده از زیرساختهای شبکه‌های رایج ارتباطی خصوصا اینترنت البته با این تفاوت که بصورت مستقل و پیوسته قابل سویچ و ارتباط با شبکه‌های بیرونی می‌باشد را یک مدل کوچک از اینترنت جهانی نامید.

اینترنت جهانی با استفاده از بستر گسترده زیرساختی خود توانسته ارتباطات را برای برای میلیونها نفر در سطح دنیا به ارمغان آورد.

اما بدلیل گسترده بودن و کنترل آن توسط دولتها در هر لحظه امکان کنترل و قطع ارتباط عادی برای کاربران در هر گوشه شبکه جهانی ممکن می‌باشد.

شبکه سایه با بهره گیری از تمام مزیتهای شبکه جهانی اینترنت , همه مزایای آن را در شبکه ای با ابعاد کوچکتر ولی ایمن تر پیاده سازی می‌نماید.

استفاده و پوشش کامل تجهیزات ارتباطی و کاربری و مدیریت آسان در کنار حفظ حریم شخصی و امنیت ارتباطی کاربران شبکه می‌تواند این شبکه را به دنیای جدیدی که آرمان شهر کاربران اینترنت جهانی است تبدیل کند.

۴- نتیجه

شبکه سایه تلاش دارد تا با استفاده بطور کمی و کیفی امنیت تبادل اطلاعات را در کنار حفظ هویت کاربران آن در سرتاسر شبکه تضمین نمایید.

بوسیله گره هایی که در سرتاسر طول شبکه گسترده شده اند می‌توان به پشتوانه یک الگوریتم قوی مسیر یابی که شبکه سایه وابسته به آن است می‌توان شبکه سایه را در هر نقطه و شرایطی ایجاد و راه اندازی نمود.

شبکه سایه با توجه به ذات محرمانگی آن می‌توان پاسخگویی کلیه نیازمندهای ارتباطی کاربران باشد.

استفاده از پروتکلها و ابزارهای متنوع ارتباطی از جمله نرم ابزارهای پیام رسان و پست الکترونیک که امروزه پر کاربرد ترین ابزارهای عصر حاظر هستند همگی قابل پیاده سازی و بهره گیری در بستر این شبکه گردند.

امروزه با افزایشهای فشارها و کنترل‌های رژیم‌های خود کامه و استکبار بر رسانه‌های عمومی خصوصا اینترنت در راستای سانسور خبری ایجاد یک شبکهارتباطی مستقل و ایمن جهت نشر اخبار و اطلاع رسانی و مقابله با دستگاه‌ها و رسانه‌های تبلیغاتی وابسته به قدرتهای غیر مردمی به شدت احساس می‌شود.

پیاده سازی و برپایی این نوع شبکه بر خلاف سایر شبکه‌های ارتباطی توسط شهروندان عادی و غیر متخصص ممکن بوده و نیاز به تهیه تجهیزات خاص نیست و می‌توان از وسایل رایج و عمومی موجود در بازار مانند مودم‌های وایرلس یا تلفن‌های همراه جهت ایجاد گره‌های ارتباطی استفاده نمود.

این شبکه بدلیل ساختار Mesh خود دارای مدیریت متمرکز نبوده و هر گره بطور مستقل بخشی از کل شبکه خواهد بود.

هر چه تعداد گره‌ها در شبکه بیشتر باشد امنیت ساختار نیز به مراتب بیشتر بوده و کربران از امنیت بیشتری برخورد خواهند بود.

علاوه بر این این نوع شبکه بدلیل غیر متمرکز بودن گره‌ها هر گز قابل مسدود سازی نبوده و شبکه حتی با ۲ گره نیز به کار خود ادامه خواهد داد و در صورت مسدود شدن یک یا چند گره در شبکه و یا ایجاد گره جعلی جهت تزریق ترافیک مزاحم توسط مهاجمین ترافیک ارتباطی بصورت هوشمند از گره‌های با ترافیک کمتر مسیر دهی خواهد شد.

با تحلیل وضعیت و شرایط خاص سیاسی جهان خصوصا در منطقه خاورمیانه و شمال آفریقا و اتحادیه اروپا و آمریکا می‌توان پیشبینی کرد که جنبشهای آزادی بخش به زودی تلاش‌های خود را برای راه اندازی این نوع مترقی از شبکه‌های ایمن ارتباطی را آغاز خواهند نمود.

سپاسگزاری

از دوست و همکار عزیز خود سرکار خانم زهرا فاطمی که بنده را در تحقیقات و ترجمه دقیق منابع مکتوب و شفاهی انگلیسی یاری کردند صادقانه تشکر و سپاسگزاری می‌نمایم.

مراجع

[1] Anonymity bibliography. http://freehaven.net/anonbib/.

[2] S. Adler, *\The Slashdot eect: an analysis of three Internet publications," Linux Gazette issue 38, March* ۲۰۰۹

[3] J. Camenisch and A. Lysyanskaya. A formal treatment of onion routing. In CRYPTO, pages 169–187, 2005.

[4] American National Standards Institute, American National Standard X9.30.2- 1997: Public Key Cryptography for the Financial Services Industry - Part 2: The Secure Hash Algorithm (SHA-1) (1997)

[5] I. Goldberg. On the security of the Tor authentication protocol. In Privacy Enhancing Technologies, 2006.

[6] P. Boucher, A. Shostack, and I. Goldberg. Freedom systems 2.0 architecture. White paper, Zero Knowledge Systems, Inc., 2000.Frankel, David S., *Model Driven Architecture: Applying MDA to Enterprise Computing*,

[7] P. Syverson, G. Tsudik, M. Reed, and C. Landwehr. Towards an analysis of onion routing security. In Designing Privacy Enhancing Technologies, pages 96–114, 2000.

This page is intentionally left blank