پردیس علوم
دانشکده ریاضی، آمار و علوم کامپیوتر

پنهان‌نگاری داده در متن فارسی

نگارنده

سهام الدین بهرامی نسب

استاد راهنما

دکتر هدیه ساجدی

پایان‌نامه برای دریافت درجه کارشناسی
در رشته علوم کامپیوتر

زمستان ۹۶

چکیده

پنهان‌نگاری، هنر و علم اطلاعات مخفی شده است. هدف پنهان‌نگاری، ارتباط مخفیانه به منظور پنهان‌سازی وجود یک پیام از چشم‌های مزاحم است. الگوریتم‌های پنهان‌نگاری مختلفی با استفاده از متن و صدا و تصویر به عنوان داده‌ی پوششی ارایه شده اند. با این حال، استفاده از متن برای این کار نسبتا سخت‌تر از انواع دیگر داده است. از جمله دلایل سخت‌تر بودن استفاده از متن، می‌توان به کم بودن اطلاعات حشو و اضافی در متن جهت پنهان‌سازی اطلاعات در آن‌ها اشاره کرد. در این پروژه به بیان برخی کارهای انجام شده در زمینه‌ی پنهان‌نگاری در متن، بررسی و تحلیل چند روش برای پنهان‌نگاری اطلاعات در متن فارسی و بهبود این روش‌ها می‌پردازیم. مهم‌ترین روش بررسی شده در این پروژه، استفاده از نیم‌فاصله است. نیم‌فاصله، یک جداکننده به طول صفر است که مانع از چسبیدن دو حرف از الفبای فارسی به یکدیگر می‌شود. با استفاده از خاصیت نامریی بودن نیم‌فاصله و قرار دادن آن در محل‌های مناسب، روشی برای پنهان‌سازی و انتقال اطلاعات معرفی می‌کنیم.

پیش‌گفتار

پنهان‌نگاری ($Steganography$)، تمرین پنهان کردن یک فایل، پیام، عکس یا تصویر در یک فایل، پیام، عکس یا تصویر دیگر است. $Steganography$ از ترکیب دو واژه‌ی یونانی $steganos$ به معنی مخفی‌شده و محافظت‌شده و $graphein$ به معنی نوشتن تشکیل شده است.

اولین استفاده‌ی ثبت شده از عبارت پنهان‌نگاری، در کتاب $Stegangraphia$، نوشته شده توسط $JohannesTrithemius$ در سال ۱۴۹۹ و با موضوع رمزنگاری و پنهان‌نگاری است. به طور کلی، پیام مخفی شده به فرم بخشی از چیزی دیگر است. مانند عکس، مقاله، لیست خرید و هر نوع داده‌ی پوششی دیگر. برای مثال، پیام ممکن است با جوهر نامریی بین خطوط مریی یک نامه‌ی خصوصی نوشته شده باشد.

تنها مزیت پنهان‌نگاری نسبت به رمزنگاری، این است که پیامی که قصد انتقال مخفیانه‌ی آن را داریم، توجه خاصی را جلب نمی‌کند و باعث پیشگیری از بررسی موشکافانه می‌شود. پیام‌های کدگذاری شده که پنهان‌نگاری نشده اند، هر چقدر که امنیت بالایی داشته باشند، باز هم توجه شخص ثالث را جلب می‌کنند و در کشورهایی که کد کردن اطلاعات خلاف قوانین است، به راحتی جرم شناخته می‌شوند.

رمزنگاری تنها به حفاظت از محتوای پیام فرستاده شده می‌پردازد. این در حالی است که پنهان‌نگاری، پیش از حفظ محتوا، به مخفی کردن وجود رمز در یک پیام اهمیت می‌دهد.

اولین استفاده ثبت شده از پنهان‌نگاری با ۴۴۰ سال پیش از میلاد برمی‌گردد. هرودوت ($Herodotus$) مورخ یونانی، در کتاب معروف خود یعنی تاریخ هرودوت (اولین کتاب تاریخ جهان) به دو مورد استفاده از پنهان‌نگاری اشاره کرده است. $Histiaeus$ پیامی برای خدمتکارش $Aristagoras$ می‌فرستد. به این

صورت که موهای مورد اعتماد ترین برده اش را تراشیده، پیام را روی پوست سر او نوشته و پس از رشد مجدد مو ها، برده را نزد خدمتکارش می‌فرستد. به علاوه، $Demaratus$ با نوشتن پیامی پشت چوب یک لوح مومی قبل از قرار دادن موم روی چوب، به یونان درباره‌ی حمله‌ی احتمالی هشدار می‌دهد.

در فصل اول این پروژه به بیان مفاهیم مقدماتی و تعاریف علم پنهان‌نگاری می‌پردازیم. در فصل دوم، فعالیت‌های عمده‌ی انجام شده در زمینه‌ی پنهان‌نگاری را بیان کرده و مزایا و معایب هر کدام را مختصرا بیان می‌کنیم. در فصل سوم، روشی مخصوص پنهان‌نگاری در زبان فارسی بیان کرده و با الگوریتم‌های متفاوت به بهبود آن می‌پردازیم.

فهرست مطالب

فصل ۱
مقدمه

پنهان‌نگاری $(Steganography)$، روشی برای مخفی کردن اطلاعات ارزشمند در داده‌ی عمومی و کم‌اهمیت است؛ به طوری که شخص ثالث متوجه وجود رمز در داده‌ی عمومی نشود. مانند رمزنگاری، پنهان‌نگاری نیز برای محافظت از اطلاعات محرمانه به کار گرفته می‌شود. با این تفاوت که در رمزنگاری، اطلاعات کدگذاری شده به راحتی باعث جلب توجه می‌شود و تمرکز تنها روی غیر قابل فهم کردن رمز برای حمله کننده است. اما در پنهان‌نگاری، پیش از پیچیده کردن رمز، سعی در مخفی کردن وجود آن می‌شود.
اصطلاحات اصلی در مفهوم پنهان‌نگاری عبارتند از:

- رمز $(plaintext)$: اطلاعات اصلی که قرار است مخفیانه منتقل شود.
- داده‌ی پوششی $(covertext)$: اطلاعاتی که قرار است رمز درون آن مخفی گردد.
- داده‌ی نهایی $(stegotext)$: محتوای به دست آمده از مخفی کردن رمز در داده‌ی پوششی.

پنهان‌نگاری، هنر مخفی کردن اطلاعات در حامل به ظاهر بی‌خطر و کم‌اهمیت، بدون جلب توجه به جابه‌جایی اطلاعات مخفی می‌باشد. در مقابل، هنر کشف و استخراج رمز از داده‌ی پوششی، پنهان شکنی $(Steganalysis)$ نام دارد. هدف اصلی پنهان شکنی، ابتدا شناسایی وجود اطلاعات مخفی و سپس شناسایی محل‌های کلیدی برای جست و جوی این اطلاعات است [۱].

ویژگی‌های اصلی مورد انتظار از یک الگوریتم پنهان‌نگاری عبارتند از ظرفیت بالا برای مخفی کردن رمز، نا مریی بودن رمز (عدم تغییر ظاهر قابل توجه داده‌ی پوششی)، غیر قابل تشخیص بودن، توانایی حفظ و نگهداری رمز، توانایی مقاومت در برابر دخالت و تغییر رمز، و مستقل بودن از داده‌ی پوششی اولیه می‌باشند [۲]. اگرچه برخی از این ویژگی‌ها با یکدیگر تداخل دارند و در نتیجه یک الگوریتم نمی‌تواند تمامی موارد گفته شده را به خوبی پوشش دهد. برای مثال، استفاده از الگوریتمی که ظرفیت پذیرش رمز بالایی دارد، احتمال شناسایی وجود رمز را بالا می‌برد. فرد باید با توجه به نیازها و هدف خود، الگوریتم مناسب را انتخاب کند.

فایل‌های صوتی و تصویری رسانه‌های محبوبی برای پنهان‌نگاری هستند. از سوی دیگر، متن با توجه به در دسترس بودن و حجم کمتر نسبت به فایل صوتی و تصویری، حوزه‌ی بسیار مناسبی برای پنهان‌نگاری می‌باشد.

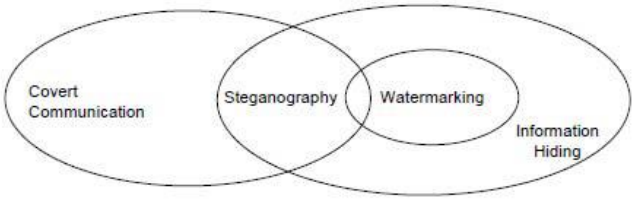
پنهان‌نگاری در متن به عنوان سخت ترین نوع پنهان‌نگاری در نظر گرفته می‌شود. مهم ترین دلیل این امر، کم بودن داده‌ی حشو و اضافه در متن در مقایسه با صدا و تصویر است [۳].

ساختار یک فایل متنی، دقیقاً همان چیزیست که می‌بینیم. این در حالی است که در سایر انواع فایل ها، برای مثال در یک عکس، ساختار آن با چیزی که می‌بینیم بسیار متفاوت است. در چنین ساختار هایی با استفاده از این ویژگی می‌توان اطلاعات را با تغییر در ساختار فایل مخفی کرد؛ در حالی که تغییر قابل ملاحظه‌ای در ظاهر فایل رخ ندهد.

ترکیب پنهان‌نگاری با سایر روش‌های حفظ اطلاعات، مانند رمزنگاری، باعث پیشرفت چشمگیر در حوزه‌ی حفظ امنیت اطلاعات در دنیای امروز گردیده است. از دیگر کاربردهای پنهان‌نگاری می‌توان به محافظت از کپی‌ـ رایت و جلوگیری از جعل فایل‌های الکترونیکی اشاره کرد [۴].

بر خلاف سایر رسانه‌ها مانند صدا، تصویر و فیلم، استفاده از متن برای پنهان‌نگاری از زمان‌های بسیار قدیم رایج بوده است. حتی پس ار اختراع دستگاه چاپ، بسیاری از کتاب‌ها و فایل ها، همچنان به صورت فقط متن باقی ماندند. حتی امروزه نیز استفاده از متن به صدا و تصویر ترجیح داده می‌شود.از دلایل آن، کم حجم بودن متن و ارزان بودن چاپ آن است.

ایده‌ی کلی مخفی کردن اطلاعات در محتوای دیجیتال حوزه‌ی وسیع تری



شکل ۱.۱: رابطه‌ی پنهان‌نگاری با سایر زمینه‌های مرتبط

از پنهان‌نگاری است.

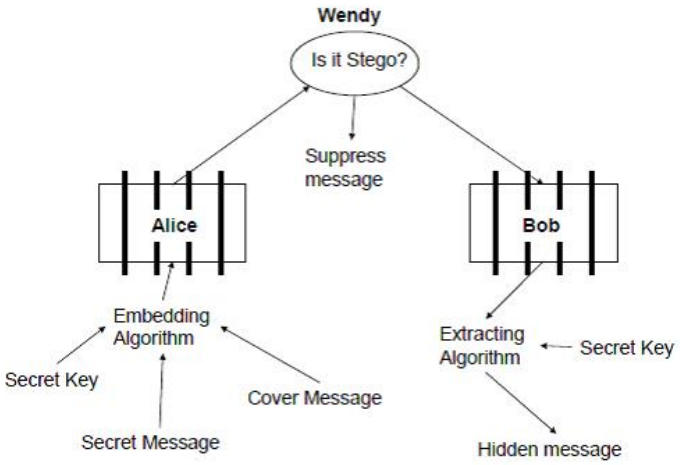
جهت مقابله با حمله‌های پنهان شکنی، الگوریتم هایی ارایه شده است که سعی میکنند اطلاعات آماری‌ای را که در طور فرایند پنهان‌نگاری تحریف شده اند و ممکن است از آنها در حمله‌ها استفاده شود، بازسازی کنند. به مجموعه‌ی این تکنیک ها، مخفی‌سازی اطلاعات $(information - hiding)$ می‌گویند. نهان‌نگاری $(digital-watermarking)$ یک حالت خاص مخفی‌سازی اطلاعات است.

نهان‌نگاری، پروسه‌ی جاسازی اطلاعات در محتوای چندرسانه ای دیجیتال است. به طوری که این جاسازی، به منظور رسیدن به اهدافی همچون کنترل و جلوگیری از جعل، برگشت پذیر یا قابل شناسایی باشد.

تفاوت کلیدی نهان‌نگاری و مخفی‌سازی اطلاعات، عدم حضور یک مهاجم فعال است. در نهان‌نگاری هایی مثل حفاظت از کپی‌ـرایت یا کدگذاری اسکناس ها، همیشه یک مهاجم به دنبال جعل، حذف یا خراب کردن رمز است. اما در مخفی‌سازی چون پاک کردن رمز کاری بی ارزش است، پس چنین مهاجمی نیز وجود ندارد.

بر خلاف دو روش یاد شده، هدف اصلی پنهان‌نگاری، برقراری یک ارتباط امن در شرایط کاملا غیر قابل تشخیص می‌باشد.صورت مدرن پنهان‌نگاری معمولا در قالب مساله‌ی زندانی [۵] بیان می‌شود. آلیس و باب دو زندانی هستند که میخواهند برای چیدن نقشه‌ی فرار، با یکدیگر مکالمه کنند. با این حال تمام مکالمات آنها توسط یک سرپرست، وندی، آزمایش می‌شود.

در پنهان‌نگاری خالص، وندی تکنیک جاسازی رمز را نمی‌داند و این تکنیک رازی بین آلیس و باب است. با این حال در حالت کلی تصور می‌شود



شکل ۲.۱: مدل کلی پنهان نگاری

که وندی از این تکنیک مطلع است و فقط کلید استفاده از الگوریتم رازی بین دو طرف است. این اصل در رمزنگاری به اصل کیرشهف معروف است.

در بخش ۲ به بیان روش‌های مختلف پنهان‌نگاری در متن می‌پردازیم. در ادامه‌ی پروژه، با بررسی و مقایسه‌ی چند الگوریتم پنهان‌نگاری، بهترین آنها را برای استفاده در متن فارسی شناسایی می‌کنیم.

فصل ۲
فعالیت‌های مرتبط

پنهان‌نگاری در متن به طور عمده به سه دسته تقسیم می‌شود: روش‌های قالب محور $(format-based)$، تولید تصادفی و آماری $(random-and-statistical-generations)$، و زبانی $(linguistic)$. روش‌های قالب محور، با استفاده از قالب متن و تغییر آن، اطلاعات را پنهان میکنند. این روش‌ها تغییری در کلمات یا جملات نمی‌دهند؛ بنابرین آسیبی به ارزش و ماهیت متن وارد نمی‌شود. در روش‌های تولید تصادفی و آماری، به طور خودکار متن پوششی مناسبی بر اساس خواص آماری زبان ساخته می‌شود. برای ساختن متن به هر زبان، از قواعد ساختاری نمونه ای مخصوص آن زبان استفاده می‌شود. روش‌های زبانی از خواص زبان شناسی متن برای تغییر آن استفاده می‌کنند. به عبارت دیگر از ساختار زبانی یک پیام به عنوان محلی برای مخفی کردن اطلاعات استفاده می‌شود. در ادامه لیستی از فعالیت‌های عمده در زمینه‌ی پنهان‌نگاری در متن به همراه مثال هایی از حمله به آن‌ها آورده شده است.

۱.۲ استفاده از حروف مشخص در متن

روشی آماری، پیچیده و وقت گیر است. حروف خاصی از برخی کلمات مشخص انتخاب می‌شوند [۶]. برای مثال، هنگامی که اگر آخرین حرف از اولین کلمه‌ی هر پاراگراف را انتخاب کرده و آنها را کنار هم قرار دهیم، رمز تشکیل شود. حمله ای بر اساس فراوانی اولین حرف کلمات ارایه شده است [۷].

۲.۲ شیفت دادن

در متن چاپ شده، خطوط متن به صورت عمودی چند درجه شیفت داده می‌شوند تا اطلاعات را مخفی کنند [۸][۹]. در روشی مشابه، اطلاعات با شیفت دادن کلمات در راستای افقی و تغییر فاصله میان کلمات مخفی می‌شوند [۸][۱۰]. این روش برای متونی قابل قبول است که در آنها فاصله‌ی بین کلمات متفاوت است. احتمال شناسایی این روش با چشم کمتر است؛ چون تغییر فاصله بین کلمات برای پر کردن خط موردی عادی است. اگر متن بازنویسی شود یا ابزارهای شناسایی حرف (OCR) به کار گرفته شوند، اطلاعات مخفی از بین می‌رود. روش پنهان شکنی برای این مورد در [۱۱] آمده است.

۳.۲ استفاده از علامت‌های نشانه گذاری

با جایگذاری مناسب علامت هایی چون "." و "،" می‌توان اطلاعات را مخفی کرد [۶]. برای این روش لازم است مکان‌های قابل نشانه گذاری در متن را شناسایی کنیم.

۴.۲ استفاده از حروف مشخص در متن

روشی آماری، پیچیده و وقت‌گیر است. حروف خاصی از برخی کلمات مشخص انتخاب می‌شوند [۶]. برای مثال، هنگامی که اگر آخرین حرف از اولین کلمه‌ی هر پاراگراف را انتخاب کرده و آنها را کنار هم قرار دهیم، رمز تشکیل شود. حمله ای بر اساس فراوانی اولین حرف کلمات ارایه شده است [۷].

۵.۲ استفاده از کلمات جایگزین

با انتخاب چند کلمه‌ی مشخص و استفلده از معانی مختلف آنها در متن می‌توان اطلاعات را مخفی کرد [۹][۱۲]. مزیت عمده‌ی این روش، حفظ اطلاعات در صورت بازنویسی و استفاده از OCR است. با این حال این

روش ممکن است گاهی اوقات معنی متن را تغییر دهد. به طور مشابه، با مخفف نویسی نیز می‌توان اطلاعات را پنهان کرد. از معایب این روش، می‌توان به ظرفیت کم آن اشاره کرد [۶]. به طوری که برای پنهان‌سازی چند بیت اطلاعات به فایلی چندین کیلوبایتی نیاز داریم. عیب دیگر این روش، شناسایی راحت وجود رمز دز صورت به کار بردن حمله‌های آماری [۱۳] است.

۶.۲ ویژگی ظاهری حروف

با تغییر برخی ویژگی‌های حروف می‌توان اطلاعات را در متن پنهان کرد [۱۴]. برای مثال در الفبای لاتین می‌توان قسمت بلند حروف h، d، b و غیره را کمی کوتاه تر یا بلند تر نوشت. بازنویسی و استفاده از OCR در این روش باعث از بین رفتن اطلاعات می‌شود. به علاوه، می‌توان نقاط و علامت‌های حروف صدا دار را در برخی حروف جا با جا کرد. زبان‌های عربی، فارسی و هندی، با توجه به فراوانی حروف نقطه دار، برای این تکنیک بسیار مناسب هستند [۱۵][۱۶][۱۷][۱۸][۱۹]. کافیست نقطه یا علامت هایی چون "ـَـُـِ" را کمی بالاتر از محل استانداردشان قرار دهیم.

۷.۲ حرف فاصله

می توان در انتهای خطوط یا پاراگراف‌ها و حتی بین کلمات، فاصله اضافه کرد [۶][۲۰]. با توجه به اینکه برخی نرم افزارهای کار با متن، قاصله‌های اضافی را حذف می‌کنند، در این روش نیز احتمال از بین رفتن اطلاعات وجود دارد. در [۲۱] نیز یک روش پنهان شکنی برای مقابله با این تکنیک آورده شده است.

۸.۲ حرف امتداد دهنده

در زبان‌های فارسی و عربی، حرفی برای امتداد سایر حروف وجود دارد که می‌توان از آن برای مخفی کردن اطلاعات استفاده کرد [۲۲]. برای مثال می‌توان "سلام" را به صورت امتداد یافته، "سـلام" نوشت.

۹.۲ ویژگی‌های نرم افزار $Office$

تکنیک‌ها و الگوریتم‌های فراوانی مربوط به ویژگی‌های این نرم افزار ارایه شده است [۲۳][۲۴][۲۵]. برای مثال، می‌توان متناسب با مقدار بیتی که قرار است مخفی شود، جهت متن را در خانه‌های مختلف تغییر داد.

۱۰.۲ استفاده از ارزش جایگاهی ارقام

می‌توان با افزودن ۰ بی ارزش در سمت چپ اعداد یا روش‌های مشابه، اطلاعات را در متن‌های دارای ارقام و اعداد (مثلا فایل‌های مربوط به امور مالی یک شرکت) مخفی کرد [۲۶][۲۷].

۱۱.۲ در تکنولوژی‌های برنامه نویسی

استفاده از برخی ویژگی ها، مانند حساس نبودن حروف بزرگ و کوچک در تگ‌های HTML و موارد مشابه دیگر [۲۸][۲۹][۳۰][۳۱].

۱۲.۲ استفاده از الگوریتم‌های فشرده سازی

در [۳۲] روشی بر مبنای فشرده‌سازی LZW با ظرفیت ۷% آورده شده است. [۳۳] و [۳۴] نیز روش‌های مشابهی با الگوریتم‌های دیگر معرفی کرده‌اند.

۱۳.۲ استفاده از ابزار ریاضی

می توان با الگوریتم‌های خلاقانه، با استفاده از قضایا و فرمول‌های ریاضی اطلاعات را مخفی کرد. در [۳۵] روشی برای مخفی کردن اطلاعات و استفاده از قضیه‌ی باقیمانده‌ی چینی برای بهتر مخفی کردن اطلاعات و کاهش میزان حجمی که در اثر اعمال الگوریتم به فایل اضافه می‌شود، آورده شده است.

فصل ۳

روش‌های برگزیده

در این پروژه، به پیاده‌سازی و مقایسه‌ی چند الگوریتم می‌پردازیم

۱.۳ نیم‌فاصله

نیم فاصله یا فاصله‌ی مجازی، نویسه ای در استاندارد یونی کد است که در بعضی زبان‌ها برای مواردی که به هم نمی‌چسبند ولی فاصله‌ی مریی ندارند، به کار می‌رود. در زبان فارسی، از نیم‌فاصله برای جداسازی حروفی که قابلیت چسبندگی دارند استفاده می‌شود. در واقع نیم‌فاصله، فاصله ای به طول صفر است. با استفاده از این خاصیت می‌توان اطلاعات را پنهان کرد. روشی بدیهی برای استفاده از نیم‌فاصله ارایه کرده و سپس به بهبود آن می‌پردازیم.

در مثال‌های ارایه شده در این بخش، به منظور قابل رویت بودن تغییرات ایجاد شده در متن پوششی، علامت " _ " را جایگزین نیم‌فاصله می‌کنیم.

۱.۱.۳ روش اول

ویژگی ظاهری نیم‌فاصله باعث می‌شود هنگامی که آن را بعد از حروفی که به حرف بعد از خود نمی‌چسبند به کار بریم، هیچ تغییری در ظاهر متن صورت نگیرد. ۷ حرف از ۳۲ حرف الفبای فارسی، هیچگاه به حرف بعد از خود نمی‌چسبند. این حروف عبارتند از: ا، د، ذ، ر، ز، ژ، و

متن	رمز	داده‌ی نهایی
پنهان نگاری، روشی برای مخفی کردن اطلاعات ارزشمند در داده‌ی عمومی و کم اهمیت است.	۱۰۰۱۰۱۱۱۰۱۰۱	پنها ‌ن نگا ‌ر ‌ی، ر ‌ و ‌ شی بر ‌ ا ‌ ی مخفی کر ‌ د ‌ ن ا طلا ‌ عا ‌ ت ارزشمند در داده‌ی عمومی و کم اهمیت است.

جدول ۱.۳: مثالی از پنهان‌نگاری با روش اول

حال می‌توان دو نیم‌فاصله را معرف کد ۰ و سه نیم‌فاصله را معرف کد ۱ در نظر گرفت. (از یک نیم‌فاصله به منظور تشخیص نیم‌فاصله‌های واقعی موجود در متن اولیه استفاده نشده است). جدول ۳.۱ نمونه ای از به کار گیری این روش را نشان می‌دهد.

تحلیل روش:

با توجه به اینکه تنها تغییر ایجاد شده، اضافه شدن چند نیم‌فاصله می‌باشد، شخص ثالث نمی‌تواند با چشم متوجه وجود اطلاعات مخفی در رمز شود. به عبارت دیگر، این روش در مقابل حمله‌های چشمی دارای‌ایمنی ۱۰۰% است.

بررسی‌های آماری نشان می‌دهند ۷ حرف استفاده شده در این روش، حدودا ۳۰% از یک متن فارسی عادی (شامل تمامی حروف و علایم نگارشی) را تشکیل می‌دهند. یعنی ظرفیت این روش ۳۰% است که میزان بسیار مناسبی می‌باشد.

این روش نیازمند فرمت خاصی نیست و بنابرین در هر متنی قابل استفاده است و مستقل از داده‌ی پوششی می‌باشد.

در مقابل به بررسی معایب این روش می‌پردازیم. در ازای هر حرف رمز، باید ۲ یا ۳ حرف (نیم فاصله) به متن پوششی اضافه کنیم که بسیار زیاد می‌باشد. در واقع، حجم فایل نهایی به دست آمده توسط این روش، با حجم داده‌ی پوششی اولیه تفاوت قابل ملاحظه ای خواهد داشت. در نمونه‌ی آزمایشی، از متنی با حجم ۲۰۰۲ بایت و رمزی ۷۲ بایتی استفاده شد که

متن	رمز	داده‌ی نهایی
پنهان نگاری، روشی برای مخفی کردن اطلاعات ارزشمند در داده‌ی عمومی و کم اهمیت است.	۱۰۰۱۰۱۱۱۰۱۰۱	پنهان‌ نگاری،‌ ‌روشی‌ ‌برای‌ ‌مخفی‌ ‌کردن‌ ‌اطلاعات‌ ‌ارزشمند‌ ‌در‌ ‌داده‌ ‌ی‌ ‌عمومی‌ ‌و‌ ‌کم‌ ‌اهمیت‌ ‌است.

جدول ۲.۳: مثالی از پنهان‌نگاری با روش دوم

فایل نهایی با حجم ۲۳۶۴ بایت حاصل استفاده از این روش بود.

از طرف دیگر با توجه به این مساله که تمامی نیم‌فاصله‌های اضافه شده خارج از مکان معمول خود هستند، کشف وجود اطلاعات مخفی در متن نهایی در صورت استفاده از کامپیوتر و برنامه‌های OCR و تحلیل آماری متن، کار دشواری نخواهد بود.

مشکل دیگر این روش، نابودی کامل اطلاعات در صورت بازنویسی متن است.

۲.۱.۳ روش دوم

اضافه کردن نیم‌فاصله پس از علایم نگارشی و حرف فاصله نیز تغییری در ظاهر متن ایجاد نمی‌کند. بنابرین می‌توان با اضافه کردن تعداد مشخصی نیم‌فاصله پس از حروف و علامت‌های گفته شده، اطلاعات را در یک متن مخفی کرد. با توجه به اینکه در متن اصلی هیچ نیم‌فاصله ای پس از کاراکترهای گفته شده وجود ندارد، می‌توان یک نیم‌فاصله را معرف ۰ و دو نیم‌فاصله را معرف ۱ در نظر گرفت.

تحلیل روش:

این روش تقریبا مشابه روش اول است. به بررسی تفاوت‌های این روش و روش قبلی می‌پردازیم. کاراکترهای استفاده شده در این روش، حدودا ۲۰٪ از یک متن فارسی را تشکیل میدهند که کماکان ظرفیت بالا و قابل قبولی است.

در این روش در قبال کاهش ظرفیت پنهان‌سازی اطلاعات، میزان تغییر

متن	رمز	داده‌ی نهایی
پنهان نگاری، روشی برای مخفی کردن اطلاعات ارزشمند در داده‌ی عمومی و کم اهمیت است.	۱۰۰۱۰۱۱۱۰۱۰۱	پنها__ن_نگا_ر__ی،_ __ر__و__شی _بر__ا_ی__ مخفی کردن اطلاعات ارزشمند در داده‌ی عمومی و کم اهمیت است.

جدول ۳.۳: مثالی از پنهان‌نگاری با روش سوم

حجم را کاهش داده‌ایم. در نمونه‌ی آزمایشی مشابه روش قبل با فایلی ۲۰۰۲ بایتی و رمز ۷۲ بایتی، فایلی با حجم ۲۲۲۰ بایت حاوی اطلاعات مخفی به دست می‌آوریم.

۳.۱.۳ روش سوم

با ادغام دو روش بالا، می‌توان اطلاعات را پس از ۷ حرف الفبا و علایم روش دوم پنهان کرد که ظرفیت فوق العاده بالای ۵۰% را به ما می‌دهد.

۴.۱.۳ روش چهارم

در روش دوم از یک و دو نیم‌فاصله برای کدگذاری استفاده کردیم. با کمی دقت می‌توان متوجه شد که استفاده از صفر و یک نیم‌فاصله نیز امکان پذیر است.

در روش‌های قبلی ظرفیت بسیار بالایی داشتیم و مشکل اصلی که با آن رو به رو شدیم تغییر حجم قابل ملاحظه‌ی فایل نهایی بود. در این روش روی برطرف کردن این عیب تمرکز می‌کنیم.

تنها از فاصله‌های موجود در متن استفاده کرده و از سایر کاراکترهای مجاز گفته شده در روش‌های قبل صرف نظر می‌کنیم. در این صوریت ظرفیتی در حدود ۱۸% خواهیم داشت که کماکان قابل قبول است.

پس از هر فاصله اگر قرار بود کد ۱ را پنهان کنیم، یک نیم‌فاصله قرار

می‌دهیم و در صورتی که قصد پنهان کردن کد ۰ را داشتیم، چیزی به متن اضافه نکرده و به سراغ فاصله‌ی بعدی برای کد بعدی می‌رویم.

مشکلی که این کار برای ما ایجاد می‌کند، هنگامی است که پنهان‌سازی تمام شده است. باید فاصله هایی که در انتهای فایل پوششی اضافه آمده و به آنها نیاز پیدا نکرده‌ایم را از فاصله هایی که برای بیان کد ۰ چیزی پس از آنها اضافه نکرده‌ایم متمایز سازیم.

برای حل این مشکل کافی است به نوعی محل پایان یافتن پنهان‌سازی را به مخاطب اطلاع دهیم. به این منظور، از آخرین فاصله ای که در پنهان‌سازی استفاده می‌شود عبور کرده و پس از فاصله‌ی بعدی دو نیم‌فاصله‌ی متوالی قرار می‌دهیم. با توجه به اینکه هیچ جای دیگر متن دو نیم‌فاصله‌ی متوالی نداریم، به هدف خود یعنی جداسازی فاصله‌های استفاده نشده و فاصله‌های دارای کد ۰ رسیده‌ایم.

این روش تغییر حجم را به طور قابل توجهی کاهش می‌دهد. اما کماکان می‌توانیم آن را بهبود دهیم. رمزی را در نظر بگیرید که غالبا از ۰ تشکیل شده است. با این روش، تنها لازم است در قسمت‌های معدودی نیم‌فاصله (معرف ۱) اضافه کنیم و فایل خروجا بسیار مطلوب و شبیه فایل اولیه خواهد بود.

حال حالتی را در نظر بگیرید که بیشتر رمز را کد ۱ تشکیل داده است (مثلا ۱۱۱...۱). به وضوح در این حالت باید تغییرات بیشتری در متن ایجاد کنیم که این مساله دلخواه ما نیست. حال راه حلی برای این مشکل ارایه میکنیم.

در ابتدا رمز را بررسی کرده و تعداد ۱ ها و ۰ ها را مقایسه می‌کنیم. هرکدام تعداد بیشتری داشت، صفر نیم‌فاصله را به آن و یک نیم‌فاصله را به دیگری اختصاص می‌دهیم و مشکل بیان شده را حل می‌کنیم.

مشکل جدیدی که راه حل بالا ایجاد می‌کند، این است که مخاطب از تصمیم ما مبنی بر این که یک نیم‌فاصله (و به طور مشابه، صفر نیم‌فاصله) را به چه کدی اختصاص داده‌ایم بی‌خبر است.

این مساله را نیز با افزودن یک (صفر) نیم‌فاصله پس از فاصله‌ی اول موجود در متن برای بیان اینکه چیزی که به کد ۰ اختصاص داده‌ایم، یک (صفر) نیم‌فاصله است، بیان می‌کنیم. به عبارت دیگر، اولین فاصله را به کد اختصاص داده شده به ۰ اختصاص می‌دهیم.

تحلیل روش:

متن	رمز	داده‌ی نهایی
پنهان نگاری، روشی برای مخفی کردن اطلاعات ارزشمند در داده‌ی عمومی و کم اهمیت است.	۱۰۰۱۰۱۱۱۰۱۰۱	پنهان_نگاری، روشی_برای_مخفی کردن_اطلاعات ارزشمند در_داده_ی عمومی_و کم_اهمیت است.

جدول ۴.۳: مثالی از پنهان‌نگاری با روش چهارم

همانطور که گفته شد، این روش دارای ظرفیت حدودا ۱۸% است.تفاوت این روش با روش‌های قبل، میران تغییر حجم فایل است. به طوری که برای داده‌ی آزمایشی مشابه موارد بالا با ۲۰۰۲ بایت حجم و ۷۲ بایت رمز، فایلی ۲۰۷۸ بایتی به دست می‌آوریم و این در حالتی است که رمز داده شده، یکی از بدترین رمزهای ممکن برای این روش (برابری تقریبی تعداد ۰ و ۱) بوده است و در صورتی که رمز فرم مناسب تری داشته باشد، حجم اضافه شده به فایل اولیه بسیار کمتر از حجم رمز خام خواهد بود.

۵.۱.۳ روش پنجم

در تمامی روش‌های گفته شده، به دلیل ظرفیت بالای روش قسمت زیادی از انتهای داده‌ی پوششی بدون استفاده می‌ماند. هدف این روش ارایه‌ی راه حلی برای استفاده از این قسمت‌ها و کاهش هرچه بیشتر تغییرات ایجاد شده در متن است.

در روش قبل مشاهده کردیم که هرچه نسبت تعداد بیت‌های مغلوب به بیت‌های غالب در رمز به ۰ نزدیک تر باشد، تغییرات کمتری بین داده‌ی اولیه و داده‌ی نهایی دیده می‌شود و در واقع تعداد کارکترهای اضافه شده به متن، دقیقا ۳ تا بیشتر از تعداد بیت‌های مغلوب است. به وضوح این مقدار مستقل از طول رمز است و فقط تابعی از تعداد بیت‌های مغلوب آن است.

ایده ای که به ذهن میرسد، کاهش تعداد بیت‌های مغلوب به قیمت افزایش طول رمز است. همان طور که گفته شد، طول رمز تاثیری بر میزان تغییرات لازم در داده‌ی پوششی ندارد. از طرفی تنها محدودیتی که برای افزایش طول رمز با آن مواجهیم، وجود مقدار مناسب حرف فاصله در متن یا همان ظرفیت

مقدار	قطعه کد	کد معادل
۰	۰۰۰۰	۰۰۰۰۰۰۰۰۰۰۰۰۰۰۰۰
۱	۰۰۰۱	۰۰۰۰۰۰۰۰۰۰۰۰۰۰۰۱
۲	۰۰۱۰	۰۰۰۰۰۰۰۰۰۰۰۰۰۰۱۰
۳	۰۰۱۱	۰۰۰۰۰۰۰۰۰۰۰۰۰۱۰۰
۴	۰۱۰۰	۰۰۰۰۰۰۰۰۰۰۰۰۱۰۰۰
۵	۰۱۰۱	۰۰۰۰۰۰۰۰۰۰۰۱۰۰۰۰
۶	۰۱۱۰	۰۰۰۰۰۰۰۰۰۰۱۰۰۰۰۰
۷	۰۱۱۱	۰۰۰۰۰۰۰۰۰۱۰۰۰۰۰۰
۸	۱۰۰۰	۰۰۰۰۰۰۰۰۱۰۰۰۰۰۰۰
۹	۱۰۰۱	۰۰۰۰۰۰۱۰۰۰۰۰۰۰۰
۱۰	۱۰۱۰	۰۰۰۰۰۱۰۰۰۰۰۰۰۰۰
۱۱	۱۰۱۱	۰۰۰۰۱۰۰۰۰۰۰۰۰۰۰
۱۲	۱۱۰۰	۰۰۰۱۰۰۰۰۰۰۰۰۰۰۰
۱۳	۱۱۰۱	۰۰۱۰۰۰۰۰۰۰۰۰۰۰۰
۱۴	۱۱۱۰	۰۱۰۰۰۰۰۰۰۰۰۰۰۰۰
۱۵	۱۱۱۱	۱۰۰۰۰۰۰۰۰۰۰۰۰۰۰

جدول ۵.۳: کدهای معادل استفاده شده در الگوریتم روش پنجم

روش است که با توجه به بالا بودن ظرفیت و بی استفاده ماندن تعداد زیادی فاصله در روش‌های قبلی، مشکلی از این بابت نداریم.

یکی از روش‌های انجام این کار، استفاده از الگوریتمی برگشت پذیر برای تبدیل رمز به یک رمز جدید با تعداد بیت مغلوب کمتر است.

الگوریتم‌های زیادی با این ویژگی وجود دارند. یکی از آن ها، استفاده از مقدار باینری کد است. برای مثال می‌توانیم رمز را به قطعه‌های ۴ بیتی تقسیم کرده و هر کدام از قطعه‌ها را با ۱۵ بیت جدید به صورت جدول زیر نشان دهیم:

همانطور که مشاهده می‌شود، ۱ بیت مغلوب خواهد شد و هیچگاه تعداد ۱ها زیاد نمی‌شود (در ۵ حالت تغییر نکرده و در ۱۱ حالت کم می‌شود). پس به هدف خود می‌رسیم.

رمز اصلی را به رمز جدید تبدیل کرده و سپس رمز جدید را در متن پنهان می‌کنیم. در مورد برگشت پذیر بودن الگوریتم نیز کافیست پس از استخراج رمز جدید از متن، آن را به قسمت‌های ۱۶ بیتی تقسیم کرده و به جای هرکدام، معادلشان را قرار دهیم. ۱ همیشه بیت مغلوب است. بنابرین همیشه صفر

متن	رمز	رمز جدید	داده‌ی نهایی
پنهان نگاری، روشی برای مخفی کردن اطلاعات ارزشمند در داده‌ی عمومی و کم اهمیت است. پنهان‌نگاری، روشی برای مخفی کردن اطلاعات ارزشمند در داده‌ی عمومی و کم اهمیت است. پنهان‌نگاری، روشی برای مخفی کردن اطلاعات ارزشمند در داده‌ی عمومی و کم اهمیت است.	۱۰۰۱۰۱۱۱ ۰۱۰۱	۰۰۰۰۰۰۱۰ ۰۰۰۰۰۰۰۰ ۰۰۰۰۰۰۰۰ ۱۰۰۰۰۰۰۰ ۰۰۰۰۰۰۰۰ ۰۱۰۰۰۰	پنهان‌نگاری، روشی برای مخفی کردن اطلاعات ارزشمند در داده‌ی عمومی و کم اهمیت است. پنهان‌نگاری، روشی برای مخفی کردن اطلاعات ارزشمند در داده‌ی عمومی و کم اهمیت است. پنهان‌نگاری، روشی برای مخفی کردن اطلاعات ارزشمند در داده‌ی عمومی و کم اهمیت است.

جدول ۶.۳: مثالی از پنهان‌نگاری با روش پنجم

نیم‌فاصله به بیت ۰ اختصاص داده می‌شود. پس این روش نیازی به بیان کد معادل بیت ۰ ندارد و تنها کافیست انتهای فایل را مشخص کنیم.

تحلیل روش:

این روش طول رمز را تقریبا ۴ برابر می‌کند. پس ظرفیت ¼ میشود. به همین دلیل ممکن است پنهان‌سازی برخی داده‌ها که قبلا قادر به انجام آن بودیم، با الگوریتم جدید ممکن نباشد. که البته ما از ابتدا مشکلی در این مورد نداشتیم و هدف ارایه‌ی روش برای داده هایی بود که ظرفیت اضافه‌ی بالایی داشتند.

مقدار	قطعه کد	کد معادل
۰	۰۰۰۰	۰۰۰۰۰۰۰۰۰۰۰۰۰۰۰۰
۱	۰۰۰۱	۱
۲	۰۰۱۰	۱۰
۳	۰۰۱۱	۱۰۰
۴	۰۱۰۰	۱۰۰۰
۵	۰۱۰۱	۱۰۰۰۰
۶	۰۱۱۰	۱۰۰۰۰۰
۷	۰۱۱۱	۱۰۰۰۰۰۰
۸	۱۰۰۰	۱۰۰۰۰۰۰۰
۹	۱۰۰۱	۱۰۰۰۰۰۰۰۰
۱۰	۱۰۱۰	۱۰۰۰۰۰۰۰۰۰
۱۱	۱۰۱۱	۱۰۰۰۰۰۰۰۰۰۰
۱۲	۱۱۰۰	۱۰۰۰۰۰۰۰۰۰۰۰
۱۳	۱۱۰۱	۱۰۰۰۰۰۰۰۰۰۰۰۰
۱۴	۱۱۱۰	۱۰۰۰۰۰۰۰۰۰۰۰۰۰
۱۵	۱۱۱۱	۱۰۰۰۰۰۰۰۰۰۰۰۰۰۰

جدول ۷.۳: کدهای معادل استفاده شده در الگوریتم روش ششم

۶.۱.۳ روش ششم

به بهبود روش قبل و افزایش ظرفیت آن می‌پردازیم. در الگوریتمی که برای افزایش طول استفاده کردیم، می‌توانیم از صفرهای سمت چپ ۱ در جدول کدها صرف نظر کنیم. جدول به این حالت تغییر می‌یابد:

از طرفی لزوما تبدیل ۴ بیت به ۱۵ (حداکثر ۱۵) بیت بهترین تبدیل نیست و ممکن اسن تبدیل ۵ به ۳۱ یا ۳ به ۷ نتیجه‌ی بهتری داشته باشند. همچنین می‌توانیم از ترکیبی از این تبدیل‌ها (که همچنان بازگشت پذیر خواهد بود) استفاده کنیم.

به علاوه، ما تا به اینجا نیم‌فاصله را تنها پس از فاصله قرار می‌دادیم. اما می‌توان بدون ایجاد تغییر در الگوریتم، قبل از هر فاصله نیز نیم‌فاصله قرار داد که با این کار ظرفیت پنهان‌سازی اطلاعات ۲ برابر می‌شود.

تحلیل روش:

تغییرات ایجاد شده، ظرفیتی که به خاطر گسترش رمز از دست داده بودیم را به ما بر می‌گرداند. همچنین در صورت نیاز می‌توانیم از سایر علامت‌های

متن	رمز	رمز جدید	داده‌ی نهایی
پنهان نگاری، روشی برای مخفی کردن اطلاعات ارزشمند در داده‌ی عمومی و کم اهمیت است. پنهان‌نگاری، روشی برای مخفی کردن اطلاعات ارزشمند در داده‌ی عمومی و کم اهمیت است.	۱۰۰۱۰۱۱۱ ۰۱۰۱	۱۰۰۰۰۰۰۰ ۰۱۰۰۰۰۰۰ ۱۰۰۰۰	پنهان نگاری، روشی برای مخفی کردن اطلاعات ارزشمند در داده‌ی عمومی و کم اهمیت است. پنهان نگاری، روشی برای مخفی کردن اطلاعات ارزشمند در داده‌ی عمومی و کم اهمیت است.

جدول ۸.۳: مثالی از پنهان‌نگاری با روش ششم

نگارشی و اعداد و ۷ حرف الفبای فارسی نیز استفاده کنیم. بنابرین مشکلی از بابت ظرفیت نخواهیم داشت.

با اجرای الگوریتم بیان شده و استفاده از ترکیب تبدیل‌های ۲ به ۳ و ۶ به ۶۳ روی داده‌ی آزمایشی با حجم ۲۰۰۲ بایت و رمزی ۷۲ بایتی (شامل ۷۲ کد ۰ و ۱)، به داده‌ی پنهان‌نگاری شده با حجم ۲۰۲۸ بایت می‌رسیم. یعنی تنها ۱۱ نیم‌فاصله به متن اضافه می‌شود (به اضافه‌ی ۲ نیم‌فاصله در انتهای فایل) (هر حرف الفبای فارسی ۲ بایت است).

فصل ۴
جمع بندی

در فصل سوم، روش‌ها و الگوریتم‌های مختلفی برای پنهان‌نگاری با استفاده از نیم‌فاصله بیان و بررسی شد که هر کدام مزایا و معایب خود را داشتند. برای مثال، روش اول علی رغم برخورداری از ظرفیت بالا، نیازمند ایجاد تغییرات زیاد برای پنهان‌نگاری اطلاعات بود. در مقابل در روش پنجم، ظرفیت بالا را در ازای کاهش میزان تغییرات لازم از دست دادیم.

در این قسمت، با استفاده از یک داده‌ی آزمایشی جدید به مقایسه‌ی روش‌های ارایه شده می‌پردازیم.

فایل پوششی، قسمتی از متن یک روزنامه است.

روش	حجم فایل اولیه	حجم رمز	حجم فایل نهایی	درصد استفاده شده از متن پوششی
روش اول	۱۱۲۰۰	۱۶۰	۱۲۰۰۸	۹%
روش دوم	۱۱۲۰۰	۱۶۰	۱۱۶۸۸	۱۵%
روش سوم	۱۱۲۰۰	۱۶۰	۱۱۶۸۸	۶%
روش چهارم	۱۱۲۰۰	۱۶۰	۱۱۳۵۸	۱۵%
روش پنجم	۱۱۲۰۰	۱۶۰	۱۱۲۸۲	۶۳%
روش ششم	۱۱۲۰۰	۱۶۰	۱۱۲۴۰	۹۸%

جدول ۱.۴: مقایسه‌ی برخی ویژگی‌های روش‌های بیان شده

فصل ۵

مراجع

[۱] N. F. Johnson، Z. Duric، S. Jajodia، Steganalysis، Springer US، Boston، MA، ،۲۰۰۱ pp. .۷۶–۴۷

[۲] D. Salomon، Data Hiding in Text، Springer New York، New York، NY، ،۲۰۰۳ pp. .۲۶۷–۲۴۵

[۳] W. Bender، D. Gruhl، N. Morimoto، and A. Lu، "Techniques for data hiding"، IBM Systems Journal، vol. ،۳۵ Issues ،۴&۳ ،۱۹۹۶ pp. .۳۳۶_۳۱۳

[۴] N. F. Maxemchuk and S. Low، "Marking Text Documents"، Proceedings of the IEEE International Conference on Image Processing، Santa Barbara، CA، USA، Oct. ،۲۹_۲۶ ،۱۹۹۷ pp. .۱۶_۱۳

[۵] G. Simmons، "The prisoners problem and the subliminal channel" CRYPTO،pp. ،۶۷_۵۱ .۱۹۸۳

[۶] T. Morkel، J. H. Eloff، M. S. Olivier، An overview of image steganography، in: Proceedings of the Fifth Annual Information Security South Africa Conference (ISSA۲۰۰۵).

[۷] X. g. Sui، H. Luo، Z. l. Zhu، A steganalysis method based on the distribution of first letters of words، in: ۲۰۰۶ International Conference on Intelligent Information Hiding and Multimedia، ،۲۰۰۶ pp. ۳۷۲–۳۶۹

[۸] S. H. Low، N. F. Maxemchuk، J. T. Brassil، L. O'Gorman، Document marking and identification using both line and word shifting، in: INFOCOM .۹۵' Fourteenth Annual

Joint Conference of the IEEE Computer and Communications Societies. Bringing Information to People. Proceedings. IEEE، ،۱۹۹۵ pp. ۸۶۰—۸۵۳ vol.۲.

[۹] A. M. Alattar، O. M. Alattar، Watermarking electronic text documents containing justified paragraphs and irregular line spacing، in: Electronic Imaging ،۲۰۰۴ International Society for Optics and Photonics، ،۲۰۰۴ pp. —۶۸۵ .۶۹۵

[۱۰] Y.-W. Kim، K.-A. Moon، I.-S. Oh، A text watermarking algorithm based on word classification and inter-word space statistics، in: Proceedings of the Seventh International Conference on Document Analysis and Recognition – Volume ،۲ ICDAR ،۰۳' IEEE Computer Society، Washington، DC، USA، ،۲۰۰۳ pp. .—۷۷۵

[۱۱] L. Li، L. Huang، X. Zhao، W. Yang، Z. Chen، A statistical attack on a kind of word-shift textsteganography، in: ۲۰۰۸ International Conference on Intelligent Information Hiding and Multimedia Signal Processing، ،۲۰۰۸ pp. .۱۵۰۷—۱۵۰۳

[۱۲] M. Niimi، S. Minewaki، H. Noda، E. Kawaguchi، A framework of text-based steganography using sdform semantics model، IPSJ Journal ۴۴ .(۸)

[۱۳] Z. Yu، L. Huang، Z. Chen، L. Li، X. Zhao، Y. Zhu، Steganalysis of synonym-substitution based natural language watermarking، International Journal of Multimedia and Ubiquitous Engineering ۴ (۲) (۲۰۰۹) .۳۴—۲۱

[۱۴] K. Rabah، Steganography-the art of hiding data، Information Technology Journal ۳ (۳) (۲۰۰۴) .۲۶۹—۲۴۵

[۱۵] M. H. Shirali-Shahreza، M. Shirali-Shahreza، A new approach to persian/arabic text steganography، in: ۵th IEEE/ACIS International Conference on Computer and Information Science and ۱st IEEE/ACIS International Workshop on Component-Based Software Engineering،Software Architecture and Reuse (ICIS-COMSAR'۰۶)، ،۲۰۰۶ pp. .۳۱۵—۳۱۰

[۱۶] M. H. Shirali-Shahreza، S. Shirali-Shahreza، A robust page segmentation method for Persian/Arabic document، WSEAS Transactions on Computers ۴ (۱۱) (۲۰۰۵)

.١٦٩٨–١٦٩٢
[١٧] J. A. Memon، K. Khowaja، H. Kazi، Evaluation of steganography for urdu/arabic text.، Journal of theoretical and applied information technology (٢٠٠٥) .٢٣٧–٢٣٢
[١٨] M. A. Aabed، S. M. Awaideh، A. R. M. Elshafei، A. A. Gutub، Arabic diacritics based steganography، in: ٢٠٠٧ IEEE International Conference on Signal Processing and Communications، ،٢٠٠٧ pp. .٧٥٩–٧٥٦
[١٩] A. Gutub، Y. Elarian، S. Awaideh، A. Alvi، Arabic text steganography using multiple diacritics، in: Proceedings of the ٥th IEEE International Workshop on Signal Processing and its Applications (WoSPA'٠٨)، University of Sharjah، Sharjah، UAE، .٢٠٠٨
[٢٠] D. Huang، H. Yan، Interword distance changes represented by sine waves for watermarking text images، IEEE Transactions on Circuits and Systems for Video Technology ١١ (١٢) (٢٠٠١) .١٢٤٥–١٢٣٧
[٢١] R. H. Rose، N. Jamal، Feasibility of text visualization in text steganalysis، in: ١٣th International Conference on New Trends in Intelligent Software Methodology Tools، and Techniques، SoMeT ،٢٠١٤ IOS Press، .٢٠١٤
[٢٢] A. Gutub، M. Fattani، A novel arabic text steganography method using letter points and extensions، in: Proceedings of World Academy of Science، Engineering and Technology، Vol. ،٢١ ،٢٠٠٧ pp. .٣١–٢٨
[٢٣] M. Khairullah، A novel text steganography system using font color of the invisible characters in Microsoft word documents، in: ٢٠٠٩ Second International Conference on Computer and Electrical Engineering، Vol. ،١ ،٢٠٠٩ pp. .٤٨٤–٤٨٢
[٢٤] Y. Bin، S. Xingming، X. Lingyun، R. Zhiqiang، W. Ruizhen، Steganography in ms excel document using text-rotation technique، Information Technology Journal ١٠ (٤) (٢٠١١) .٨٩٣–٨٨٩
[٢٥] R. K. Tiwari، G. Sahoo، Microsoft excel file: A steganographic carrier file، International Journal of Digital Crime and Forensics (IJDCF) ٣ (١) (٢٠١١) .٥٢–٣٧
[٢٦] M. Khairullah، A novel text steganography system

in cricket match scorecard، International Journal of Computer Applications ۲۱ (۹) (۲۰۱۱) .۴۷–۴۳

[۲۷] M. Khairullah، A novel text steganography system in financial statements، International Journal of Database Theory and Application ۷ (۵) (۲۰۱۴) .۱۳۲–۱۲۳

[۲۸] K. Bennett، Linguistic steganography: Survey، analysis، and robustness concerns for hiding information in text، Tech. rep.، Purdue University، cERIAS TR ۱۳_۲۰۰۴.(۲۰۰۴)

[۲۹] H. Huang، J. Tan، X. Sun، L. Liu، Detection of Hidden Information in Webpage Based on Higher- Order Statistics، Springer Berlin Heidelberg، Berlin، Heidelberg، ،۲۰۰۹ pp. .۳۰۲–۲۹۳

[۳۰] H. Kabetta، B. Y. Dwiandiyanta، et al.، Information hiding in css: A secure scheme text-steganography using public key cryptosystem، Int. Journal on Cryptography and Information Security ۱ (۲۰۱۱) .۲۲–۱۳

[۳۱] Y. Bassil، A generation-based text steganography method using sql queries، International Journal of Computer Applications ۵۷ .(۱۲)

[۳۲] E. Satir، H. Isik، A compression-based text steganography method، Journal of Systems and Software ۸۵ (۱۰) (۲۰۱۲) ۲۳۸۵ – ،۲۳۹۴ automated Software Evolution.

[۳۳] E. Satir، H. Isik، A huffman compression based text steganography method، Multimedia Tools and Applications ۷۰ (۳) (۲۰۱۴) .۲۱۱۰–۲۰۸۵

[۳۴] A. Malik، G. Sikka، H. K. Verma، A high capacity text steganography scheme based on LZW compression and color coding، Engineering Science and Technology، an International Journal ۲۰ (۱) (۲۰۱۷) ۷۲ – .۷۹

[۳۵] S. G. R. Ekodeck، R. Ndoundam، PDF steganography based on chinese remainder theorem، Journal of Information Security and Applications ۲۹ (۲۰۱۶) ۱ – .۱۵

واژه‌نامه

رمز ..... Plaintext
داده‌ی پوششی ..... Covertext
رمزنگاری ..... Cryptography
پنهان‌نگاری ..... Steganography
پنهان‌شکنی ..... Steganalysis

Abstract

Steganography is the art and science of covered or hidden writing. The purpose of steganography is covert communication to hide the existence of a message from the prying eyes. Digital Steganography algorithms have been developed by using texts، images and audio as the cover media. However، using text as the target medium is relatively difficult as compared to the other target media، because of the lack of available redundant information in a text file. In this project، we introduce an approach for text steganography using Zero-width non-joiner (ZWNJ) and try to improve it. ZWNJ is a non-printing character used in the computerization of writing systems that make use of ligatures. When placed between two characters that would otherwise be connected into a ligature، a ZWNJ causes them to be printed in their final and initial forms، respectively. This is also an effect of a space character، but a ZWNJ is used when it is desirable to keep the words closer together or to connect a word with its morpheme. We use the fact that ZWNJ is invisible to eyes and hide a message in a cover text by putting some ZWNJ in the right place.

Faculty of Science
School of Mathematics، Statistics and Computer Science

Steganography in Persian text

Author:

Saham Bahrami

Supervisor:

Dr. Hedieh Sajedi

A thesis submitted in fulfillment of the requirements
for the degree of B.Sc in
Computer Science

Spring 2018